SBIR-09.05--8988A

NASA CR-188375

Transitions Research Corporation *NASA Phase II SBIR Final Report* *December 1994*

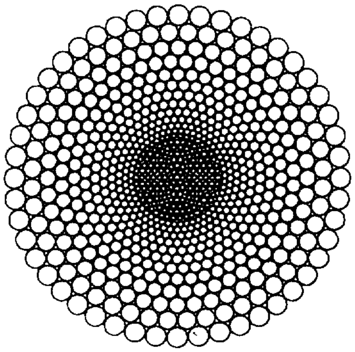
LOG-POLAR BINOCULAR VISION SYSTEM

IN-74
51474

Final Report

SBIR Phase II Contract # NAS 9-18637

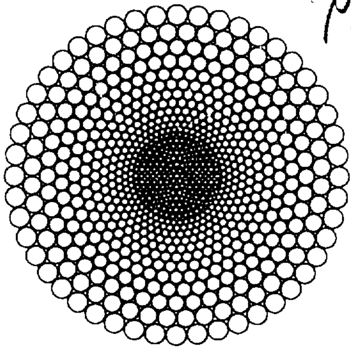
P, 91



December 1994

Transitions Research Corporation
Shelter Rock Lane
Danbury, CT., 06810
(203) 798-8988

Carl F. R. Weiman, Ph. D.,
Principal Investigator
Timothy E. Fisher, NASA JSC Contracting Officer Technical Representative



Abstract

The objective of this research is to develop and implement new techniques for real time stereo vision for robots, based on active binocular vision using log-polar pixel layout (shown above) and novel Gabor filter processing. The active vision platform is a high performance servo controlled binocular mount designed and built under this project. It emulates the articulation and speed of the human eye-head system. High performance processing is based on local Gabor filters embedded in global log-polar image plane geometry. This choice of geometry and processing, developed from first principles, closely resembles the structure and function of the primate retina and visual cortex.

Binocular stereo vision is demonstrated on an autonomous mobile robot equipped with active binocular vision. Using real time visual guidance, the robot can perceive 3-D environments and maneuver according to its perceptions.

The new approach to vision relieves the computational bottlenecks of conventional machine vision, and provides robust sensory information which is well suited to real time maneuvering and manipulation. Space applications include eye-hand coordination for manipulators and high precision passive sensing for docking. Commercial applications include 3-D visual telepresence for remote inspection and repair in nuclear and hazardous environments, visual autopilot for autonomous vehicles, and real time vision for robot arms in unstructured manipulation environments. The latter includes semi-autonomous aides for the elderly and handicapped.

(NASA-CR-188375) LOG-POLAR BINOCULAR VISION SYSTEM Final Report (Transition Research) 91 p

N95-3077

Unclas

G3/74 0051474

TABLE OF CONTENTS

Section *Page*

PRECEDING PAGE BLANK NOT FILMED

1.0 OVERVIEW

Subject

This study developed binocular stereo techniques based on log-polar retinas and a new discrete Gabor filter. Some of these techniques were applied in real time active vision algorithms operating on an autonomous mobile robot. The term "active vision" refers to camera servos controlled by visual feedback. Active vision is a rapidly emerging field of research at the frontiers of robot vision. Log-polar imaging allocates high resolution to the center of the field of view and is therefore intimately dependent on active cameras to steer the focus of attention. Gabor filters can be used as 2-D signal processing elements whose applications include measurements of optic flow, local texture, and binocular disparity to subpixel precision.

Need and Dual Use

Many of NASA's space robotic tasks require 3-D spatial perception. EVA retriever, satellite servicing, autonomous docking and manipulation are examples. Energy emitting sensors such as radar and LIDAR consume scarce power and have limited range. Radar has lower resolution than visual data, and LIDAR takes longer to scan per frame than video images at the same resolution. Thus, for high precision real-time tasks, passive (non-emitting) vision is superior to alternative sensors. Commercial applications encompass all of robotics, including new niches which are not yet occupied because of the limitations of conventional machine vision techniques and architectures. Applications include intelligent mobile vehicles in unstructured environments, i.e., security, transport, repair and autopilot applications, and ultimately, the household robot. Close relatives of the latter include mechanical servants to aid the handicapped and the aged, potentially enormous markets.

Approach

High resolution imagery is a valuable but costly resource which must be efficiently allocated in any robot vision system. Log-polar image plane pixel arrays concentrate this resource dead center in the field of view, linearly decreasing resolution from the center. Pixel count is thereby reduced by nearly two orders of magnitude, reducing the total computation rate by the same ratio. This speedup is of paramount importance for real time control; servo lags would otherwise seriously degrade performance. Active vision aims the highest resolution viewfield at the focus of attention; binocular coordination precisely measures range and surface configuration.

The choice of log-polar image plane coordinates has important geometric consequences for 3-D stereo, in addition to the computational speedup cited in the preceding paragraph. Pixels from the 2-D retina divide the visible environment into 3-D cones by projection. The intersections of these cones from two (binocular) image planes divide the environment into compartments called "voxels" (volume cells).

The cluster of high resolution voxels at the intersection of projected centers of log-polar retinas defines a high precision zone of attention which can be steered in three dimensions like the intersection of a pair of crossed spotlights converging on a target.

The visual attention process is driven by locking on to high-contrast features which can be quantified as outputs of local filters. We selected Gabor filters because of their optimal joint uncertainty reduction in spatial and frequency domains. Our goal was to develop a discrete Gabor filter which would be practical for real time computation.

In our analysis and design of visual mechanisms, we selected the software tool *Mathematica*® to derive equations, implement algorithms and generate graphics.

Our proof-of-concept plan was to design and build a high performance servo controlled binocular vision platform to test binocular stereo algorithms in real time. The platform was to be mounted on a mobile robot platform for navigation experiments.

Results

This research successfully demonstrated the superb real time performance of an active binocular vision system. The working prototype mimics many of the properties of human vision, and opens the door to a new approach to real time robot vision, with commercial applications in service robots and NASA applications in vision guided space manipulation and maneuvering.

A major innovation in this research was the design of a new discrete Gabor filter. This filter is a high performance 2-D signal processing operator whose application is far more general than the measurement of binocular disparity. It can be used for optic flow, texture characterization, pattern recognition, shape from shading, and many other local-operator based vision computations. Its computational function resembles that of the simple and complex cells in the primary visual cortex. In binocular applications, our experiments showed that the new discrete Gabor filter performed sub-pixel disparity measurements robustly and sensitively using less computation than required by other methods, such as correlation. An exposition of the new discrete Gabor filter was published in *Proceedings of the SPIE Conference on Wavelet Applications* in Orlando, Spring 1994 [Weiman 1994a]. The Principal Investigator received a NASA Certificate of Recognition for this work, which will also be reported in an upcoming issue of NASA Tech Briefs.

The design and construction of a new high-performance servo-controlled binocular camera platform in this project represents a major advance in the development of robotic devices. This new platform coordinates the motion of microminiature binocular cameras at significantly higher speed and precision, for such a small package, than any commercially available platform. Steel band pulleys transmit large torques to the camera mounts without backlash. Speeds in excess of 1,000° /second were achieved. Servo control was implemented using a high performance 8-axis industrial controller.

The system performed very smoothly, quickly, and accurately. Potential NASA applications for this camera platform include remote 3-D vision for virtual reality and autonomous binocular stereo for manipulators and docking.

The binocular camera platform is mounted on a self-contained mobile robot base which also carries the PC/AT which houses the controller, frame grabber and image processor. Images were captured and mapped to log-polar coordinates. Encoder readings and image analysis outputs are used as error signals to steer the cameras. No external processors or power sources are necessary, eliminating the need for umbilicals.

Experiments included simple tracking tasks, and navigation guidance via 3-D inference from binocular viewing. The camera platform and algorithms performed superbly in these experiments.

The geometric analysis of log-polar retinas led to a "quantum geometry" which yields general design principles for filling the singularity at the center of the field of view with uniform high resolution pixels. Application of the rules of this geometry led to crystallographic and polar tilings for the high resolution central region (called the fovea after its biological counterpart), smoothly blending with the log-polar periphery. A variety of useful foveal designs were generated and plotted.

Geometric analysis of the 3-D intersection of log-polar pixel projections showed some clear advantages for log-polar retinas as opposed to Cartesian retinas for binocular stereo vision. The cluster of high resolution voxels at the intersection of the two view axes provides a superb mechanism for 3-D focus of attention. With Cartesian image planes, there is no such center and resolution is wasted at the periphery of the zone of binocular registration. Geometric cross-sections of the log-polar voxels using *Mathematica* provide a clear picture of the high resolution zone in the center of the field of view.

In summary, theory led to important new principles for the design of real time robot vision systems, and some of these principles were successfully embodied in a real time mobile robot running under the control of input from an active binocular vision system.

Spin-offs and Future Work

This project proved the viability of the new active binocular vision techniques based on log-polar retinas and discrete Gabor filters. Their significant advantages over alternative methods opens the door to practical real time robot vision for navigation and manipulation. Specific results of this work have immediate implications, as follows.

The most far reaching result of this research was the design of a new discrete Gabor filter. Its generally useful properties and high efficiency make it a candidate for the central element of a new image processing architecture. The close resemblance of the Gabor filter to neural units in the visual cortex suggest that the name "silicon cortex" is appropriate for such architectures.

The principal investigator is opening dialogs with other researchers as potential collaborators on the design of a multi-GIPS (Giga- or 10^9 instructions per second) processor for video compression and real time stereo and pattern recognition. One such dialog led to a Phase I STTR award (1994) from BMDO to explore feasibility of such an architecture.

The binocular camera motion platform served as a Mark 1 prototype for a research product, namely, an active vision mount whose Mark 2 version, developed under National Science Foundation funding, is now sold by TRC to Universities and research institutions. The robotic system delivered to JSC from this research will provide NASA with a flexible test bed to develop visually guided navigation, tracking and docking techniques. And finally, TRC will employ the techniques of visual guidance in its next generation of mobile robots in the service sector.

The fovea geometries we developed provide designs to improve upon the layout of silicon retinas such as the world's first fabricated by IMEC [Sandini and Dario, 1989, Van der Spiegel et al, 1989] in Belgium, or the later model developed by Carver Mead with Eric Schwartz [Rojer, A. S. and Schwartz, E. L., 1990, Mead 1989]. The geometric principles of binocular fields of view lead directly to control algorithms which can be applied in the rapidly emerging field of active vision.

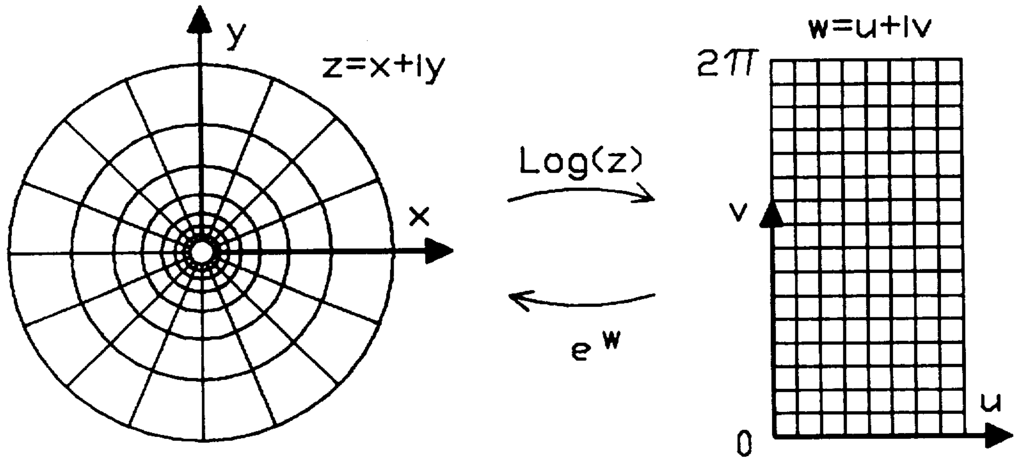
Guide to Contents of this Report

The remainder of this report details the results of theoretical analysis and experimental research. Section 2 below describes the design of foveas for log-polar retinas. Section 3 analyzes 3-D stereo geometry for log-polar sensors. Section 4 describes the new discrete Gabor filter and its application to disparity. Section 5 describes the design of the binocular vision platform and vehicle. Section 6 describes control algorithms and demonstration scenarios. Summary and references are found in Section 7. Page, figure, and equation numbers are prefixed by hyphenated section number, as in "Figure 3-9" for the 9th figure in section 3.

2.0 FOVEAS FOR LOG-POLAR RETINAS

Figure 2-1a illustrates an array of pixels defined by a log-polar coordinate grid. The term "log-polar" arises from the natural *(log(r), θ)* coordinate grid which differs from the familiar *(r, θ)* system of polar coordinates by taking the logarithm of the radial coordinate. This choice of coordinates is motivated by a desire to maximize both field of view and resolution while economizing on pixel count. These conflicting requirements can be reconciled by concentrating high resolution at the center of the field of view and progressively decreasing peripheral resolution. Preservation of locally isotropic pixel neighborhoods and global zoom/rotation symmetry are realized in a pattern of expanding rings of isotropic pixels whose size is proportional to distance from the center, i.e., the log-polar array.

This distribution (figure 2-1a) is succinctly expressed as the complex exponential (conformal) mapping of a Cartesian coordinate domain (figure 2-1b) as annotated in the figure. This representation simplifies formulation of geometric transformations such as rotation, zoom, and perspectivity by reducing them to elementary arithmetic (vector addition, sign reflection) on complex numbers. In vision, figure 2-1a corresponds to the image plane pixel layout, and figure 2-1b to the pixel indexed data structure in a frame buffer. Thus rays and rings of pixels map into rows and columns of data. NASA JSC [Fisher and Juday, 1988] and Transitions Research Corporation [Weiman, 1990b] have built real time hardware which performs this transformation of coordinates on video data. IMEC [Van der Spiegel et al., 1989] fabricated a CCD chip which incorporates physical pixel layout similar to that in figure 2-1a. The ultimate motivation for the designs presented below is embodiment in "silicon retina" pixel geometry, although that goal is beyond the scope of the present project.



a) Image plane pixel layout

b) Pixel index data structure

Figure 2-1. Log-polar image plane and mapping

The log-polar mapping

$$w = log(z) \tag{2-1}$$

contains an essential singularity at the origin, i.e.

$$log(0) = -\infty \tag{2-2}$$

The corresponding geometric property is that pixels become infinitesimally small and infinitely numerous at the center of the field of view in figure 2-1a, a physically unrealizable configuration. The center of the image plane must therefore deviate from the log-polar pattern. In the human retina, to be described later, this anomalous central region is called the fovea, a name which we adopt for the anomalous center of the artificial retinas whose designs are discussed at length below.

A straightforward solution for the design of the fovea, illustrated in figure 2-2, fills the hole in the center of figure 2-1a uniformly with cells whose areas equal the areas of pixels in the inner ring of log-polar pixels. This is the approach used by IMEC for the world's first log-polar chip, illustrated in figure 2-3. One benefit of this approach is that linear coordinates are used in the center, permitting traditional translation-invariant pattern recognition and uniform neighborhood size. The discontinuity in coordinates at the boundary, where *(x, y)* coordinates abruptly switch to *(log(r),* θ*)*, poses a problem for image transformations across this boundary, for example, translation and rotation.

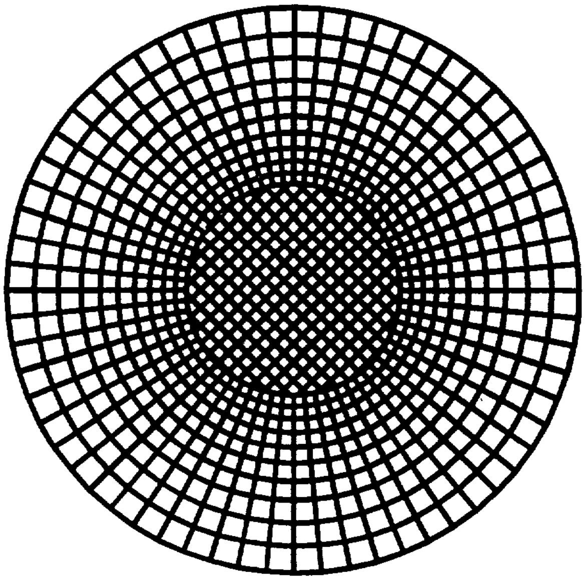


Figure 2-2. Isotropic central fill

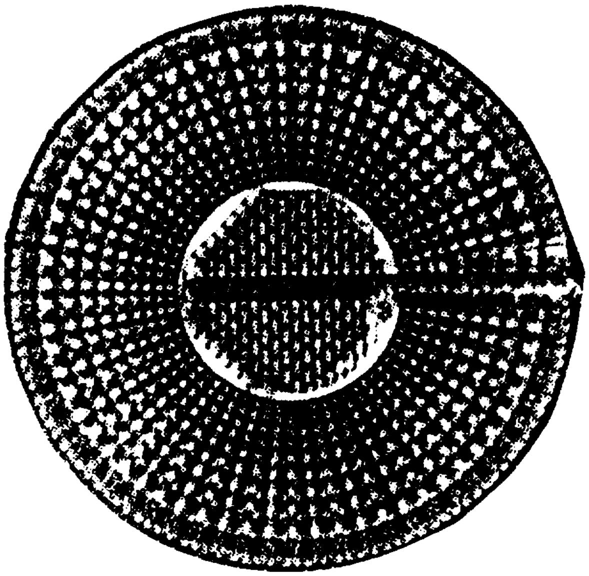


Figure 2-3. Fovea of IMEC sensor chip

2.1 Log(z+a) Retinas

Schwartz [Rojer and Schwartz, 1990] proposed a "fovea-less" solution to eliminate the singularity at the origin by using the mapping

$$log(z+a) \tag{2-3}$$

which is not singular at $z = 0$. But what if $z = -a$? Schwartz precludes this condition by restricting z to the right half plane for $log(z+a)$ i.e. $Re(z) > 0$, and using a mirror image map

$$log(z-a) \tag{2-4}$$

for the left half z-plane. This is equivalent to cutting a vertical strip of width $2{\bullet}a$ from the image plane, and moving the two half-maps of $log(z)$ to join on the y-axis, as shown in figure 2-4.

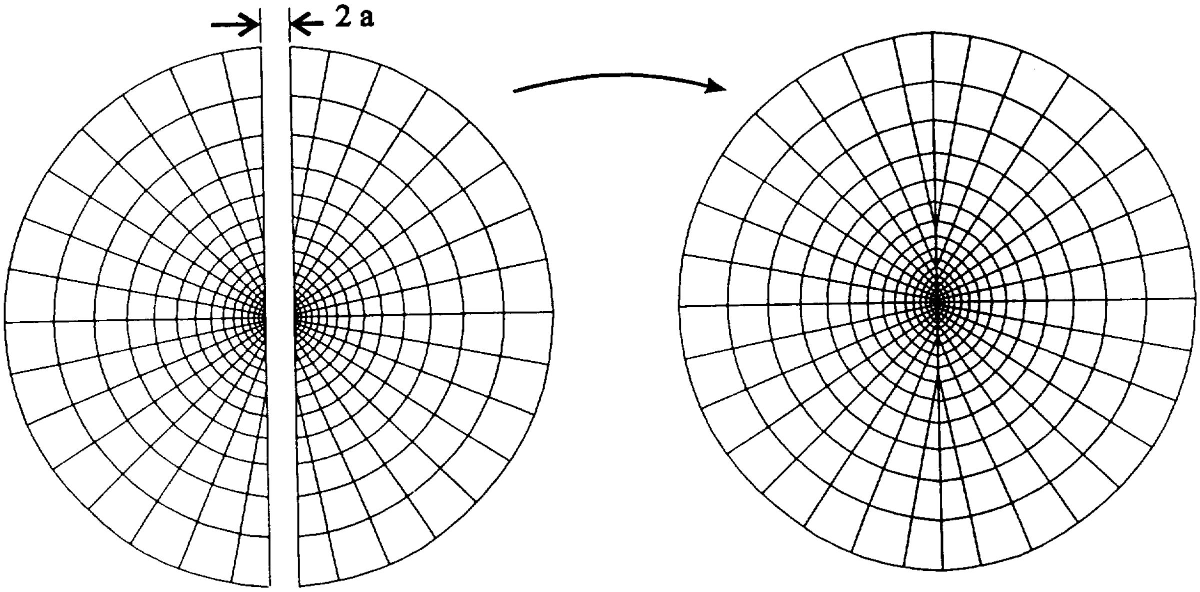


Figure 2-4. Split *log(z+a)* mapping

The construction is easily verified by examining the real and imaginary parts of the inverse mapping of equation 2-3,

$$e^{w} = z+a \tag{2-5}$$

where

$$w = u+i\,v \tag{2-6}$$

and

$$z = x+i\,y \tag{2-7}$$

whence

$$e^{w} = e^{u+i\,v} = e^{u} e^{i\,v} = e^{u}\left(cos(v) + i\,sin(v)\right) = x + a + i\,y \tag{2-8}$$

Equating real and imaginary components yields

$$x = e^{u} cos(\,v\,) - a \;\; ; \;\; y = e^{u} sin(\,v\,) \tag{2-9}$$

For u constant (verticals in the w-plane), image plane (z-plane) trajectories are circles centered at $(-\,a, 0)$. Clearly the center of the circles, $z = -\,a$, is a singularity so Schwartz stops z at $Re(z) > 0$ and takes mirror image precautions ($Re(z) < 0$) likewise for the left-half mapping (equation 2-4), yielding figure 2-4. This draconian measure splits the image plane with a slope discontinuity (90° to 180° for $0 \le \mathrm{Im}(z) \le a\sqrt{2}$) for u-level lines right down the middle, and destroys rotation and zoom symmetry.

2.2 Concentrically Inward Fovea Construction

The following solution to the fovea problem retains log-polar symmetry intact in the periphery, blending rotationally symmetric rings inward, subject to the following constraints in the fovea:

Constraint 1. Equally spaced concentric rings of pixels
Constraint 2. Equal area pixels
Constraint 3. Unit aspect ratio pixels

The general objective is to maintain rotational symmetry for continuity with the log-polar periphery, while freezing pixel size within the fovea. Some of the constraints are mildly contradictory, as we shall show. However, by prioritizing them in the order listed above, and gently compromising the lower priorities, we derive recipes for a few "best" foveas. We proceed below by quantifying pixel area at the outer boundary of the fovea, and working inward to the center with concentric rings of (nearly) equal area pixels.

Consider a log-polar retina with n pixels per ring, e.g. figure 2-1 has $n=16$ pixels per ring. Define area in units normalized to the area of pixels in the innermost ring of the log-polar region just outside the fovea, i.e. consider these as unit-area pixels. Since there are n of them per ring, the area of the innermost log-polar ring is n. Expressing the area of this annulus in terms of its inner and outer radii, r_0 and r_1 respectively, where

$$r_1 = r_0\, e^{2\pi / n} \tag{2-10}$$

yields

$$\pi\left(r_1^{\,2} - r_0^{\,2}\right) = \pi\; r_0^{\,2}\left(e^{4\pi / n} - 1\right) = n \tag{2-11}$$

whence

$$r_0 = \frac{\sqrt{n/\pi}}{\sqrt{e^{4\pi/n} - 1}} \quad . \tag{2-12}$$

Now, r_0 is in linear units whose square is unit area, the size of pixels with which we wish to populate the fovea as specified by constraint 2, above. Following constraint 3, the width of the pixel should equal its height. Both dimensions should therefore be unity. For rings of pixels as shown in figure 2-5, width and height correspond to arc length of the midline, $(r_j + r_{j+1})\pi/n$, and thickness of the ring, respectively. Their product is the area of the pixel. Now comes some "quantum geometry". The quantity r_0 is generally not an integer (equation 2-12!), but the number of rings must be. To avoid any stunted rings at either end, we divide r_0 into equal segments as nearly equal to unity as possible to define concentric ring boundaries, in accordance with constraint 1. The exception is the degenerate innermost ring of the fovea which is a disk of unit area, i.e., whose radius is

$$r_c = 1/\sqrt{\pi} \quad . \tag{2-13}$$

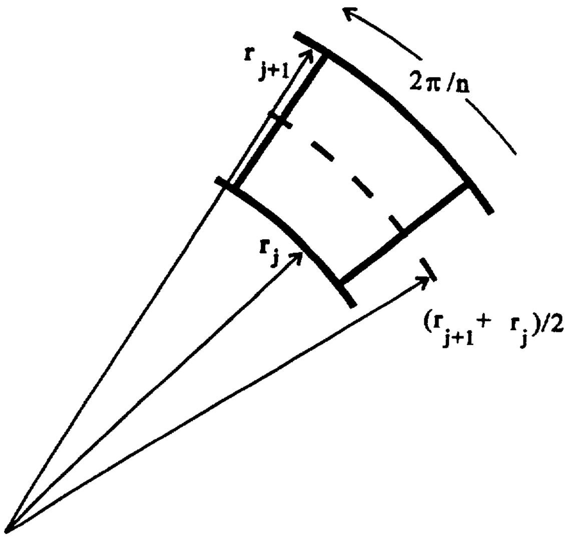


Figure 2-5. Pixels as sectors of rings

Thus, the "best" (closest to unit thickness) number of ring boundaries intervening between r_c and r_0 is

$$m = Round(r_0 - r_c) \tag{2-14}$$

and the thickness of the rings is therefore

$$h = (r_0 - r_c)/m \quad . \tag{2-15}$$

Now we have another quantum geometric constraint in the orthogonal dimension: each ring must be divided into an integer quantity of equal pixels whose area is as close as possible to unity. The inner radii of the sequence of rings is

$$r_k = r_0 + k\,h\ ;\ k = -1, -2, -3, \ldots -m \tag{2-16}$$

(Note: For inwardly built foveas we use negative indices within the fovea, positive without)

Thus the area of the k<u>th</u> ring is

$$A_k = 2\pi\left(r_0 + k\,h + \tfrac{1}{2}\right)\cdot h \tag{2-17}$$

whence the number of pixels in the k<u>th</u> ring is

$$n_k = Round(\ A_k\ ) = Round\left(\ 2\pi\left(r_0 + k\,h + \tfrac{1}{2}\right)\cdot h\ \right)\ . \tag{2-18}$$

To summarize, partition the fovea into equally spaced rings whose thickness is nearly unity, in units whose square is the area of one pixel of the first ring of log-polar pixels outside the fovea. Then divide each ring equally into sectors, each of whose area is as close to unity as possible. These two design principles quantize the distribution of rings within the fovea, and the distribution of pixels within each ring, subject to the three original constraints.

Now let us consider some examples for values of *n*, the number of rays in the log-polar periphery, which yield reasonable resolution ranges. First, we restrict the choice to even values of *n* to yield horizontally and vertically symmetric retinas. Such symmetries are consistent with a visual world incorporating horizons, gravity, and binocular epipolar planes. A further restriction to powers-of-two yields tessellations which readily admit multi-resolution pyramids in the log-polar domain, a useful but not essential characteristic.

Table 2-1 lists foveal parameters for various values of *n*, the number of pixels each ring of log-polar pixels. While most are powers of 2, we also include *n= 48* because it is the minimum size which provides visually recognizable reconstructions in image compression [Weiman 1994b]. The values of *m* and *h* in table 2-1 are computed in accordance with equations 2-14 and 2-15. Pixel count per ring, n_k, was computed from equation 2-18. Figure 2-6 illustrates some foveas which were generated using the parameters from table 2-1. In each case, the outer ring of pixels is the first ring of the log-polar periphery for the values of *n* indicated. Note the graceful transition from log-polar periphery to fovea.

Observe from table 2-1 and figure 2-6 that the number of pixels per ring decreases by "nearly" 2π. This is a consequence of the fact that a radius *r* ring (annulus) one unit thick has area $2\pi\ r$, so the difference in areas of successive rings is 2π . Quantization into equal area pixels yields integers whose difference between rings is mostly 6 but sometimes 7, a leap-year like progression whose average step is 2π. The ring sequence for $n=48$ deviates from this pattern because ring thickness (equation 2-15) differs more from unity (thinner) than the sequences for other values of n. Thus pixel count only drops 5 in two of its rings, as the quantity n_k strives to accommodate unit pixel area.

n	m	h	n_k
32	4	1.009	26, 20, 13, 7
48	7	.94	40, 34, 29, 23, 17, 12, 6
64	9	1.014	58, 52, 46, 39, 33, 26, 20, 13, 7
128	19	1.016	124, 117, 111, 104, 98, 91, 85, 78, 72, 65, 59, 52, 46, 39, 33, 26, 20, 13, 7

Table 2-1. Fovea parameters for various perimeter resolutions

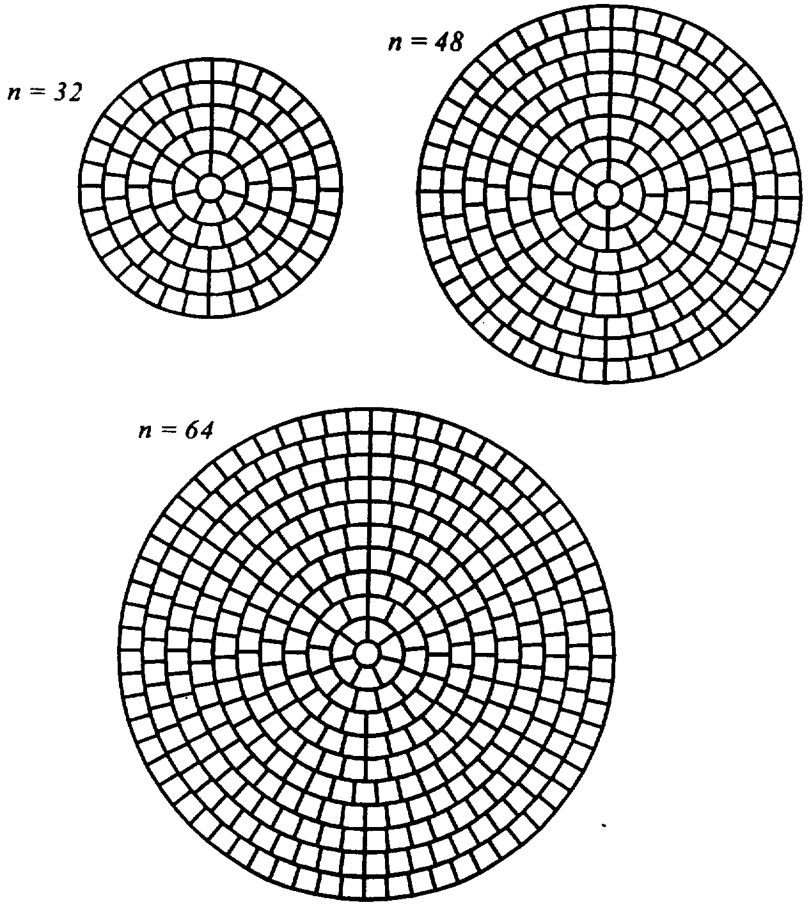


Figure 2-6. Selected foveas based on table 2-1

2.3 Concentrically Outward Fovea Construction

The constructions described in the preceding sub-section build foveas ringwise from the outside inward, preserving equal spacing between rings but compromising pixel area somewhat. While this method guarantees matching log-polar pixel count at the periphery of the fovea, it can leave awkward central patterns, e.g. 7 pixels per ring, which may be undesirable for image processing at the center of the field of view. An alternative approach builds the fovea from the inside outward, starting with central pixel(s) of specified symmetry. Guided by similar constraints as in the preceding section, i.e., concentric rings of unit area pixels with unit aspect ratios, start with a circle at the center (the degenerate 0^{th} ring in subsequent subscript notation) which is divided into n_0 equal pie-shaped segments as illustrated in figure 2-7. For segments of unit area, the radius of the circle must be

$$r(n_0) = \sqrt{n_0/\pi} \quad . \tag{2-19}$$

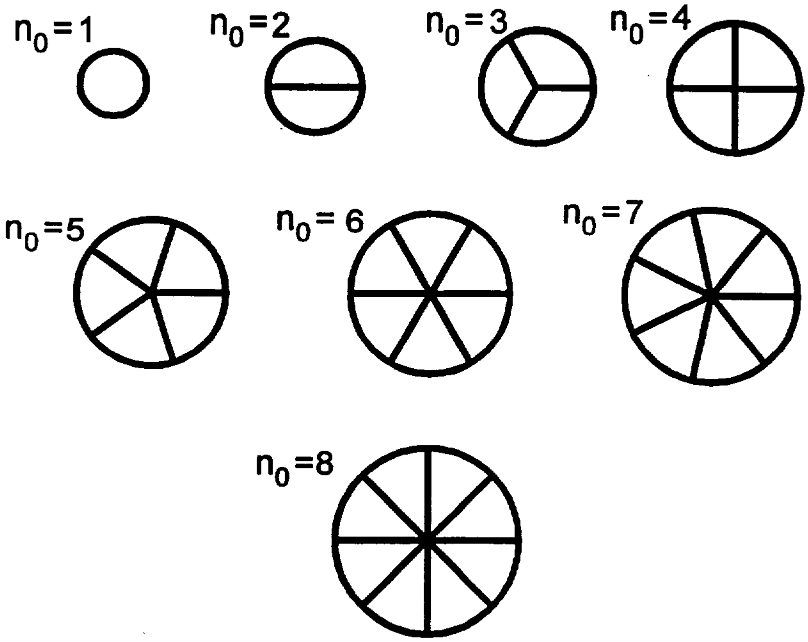


Figure 2-7 Symmetry options for central pixel(s)

Clearly, large values of n_0 would yield skinny pixels in violation of the unit aspect ratio constraint. We quantify this measure below to set reasonable limits in central pixel count.

The aspect ratio of the undivided circle ($n_0 = 1$) is unity. For larger values of n_0, segment aspect ratio can be defined as

$$a(n_0) = r(n_0) / c(n_0) \quad , \tag{2-20}$$

the radius, r_0 , divided by chord length, $c(n_0)$, where

$$c(n_0) = 2r\,sin(\pi / n_0) \quad . \tag{2-21}$$

Table 2-2 lists radii, $r(n_0)$, aspect ratios, $a(n_0)$, and reciprocals of aspect ratios (for comparable measures on either side of unity) for pie-chart segments of unit area as illustrated in figure 2-7.

n_0	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
$r(n_0)$	.56	.80	.98	1.13	1.26	1.38	1.49	1.60	2.86	3.18
$a(n_0)$	1.0	.5	.58	.71	.85	1.0	1.15	1.31	1.46	1.62
$1/a(n_0)$	1.0	2	1.73	1.41	1.18	1.0	.87	.77	.68	.62

Table 2-2. Aspect ratios of pie-chart pixels

It is apparent that reasonable aspect ratios, i.e. between .7 and its reciprocal, are restricted to values

$$n_0 = 1, 4, 5, 6, 7, 8 \tag{2-22}$$

The values $n_0 = 5, 7$ are to be rejected by virtue (vice) of their awkward symmetries. Having chosen a particular value of n_0 for the center, the strategy is to build successive rings of unit thickness

$$r_{j+1} = r_j + 1 \quad , \quad j = 0, 1, 2, \ldots \tag{2-23}$$

and divide them into whole numbers of pixels, each of whose area is as close to unity as possible. Since ring area is

$$A_j = 2\pi\left(r_j - \frac{1}{2}\right) \quad , \tag{2-24}$$

the number of pixels per ring is

$$n_j = Round\left(A_j\right) \quad . \tag{2-25}$$

Now, given n_0 as the seed for r_0 (equation 2-19), equations 2-23 and 2-25 yield the number of pixels per ring as

$$n_j = Round\left(2\pi \, \left(r(n_0) + j - \tfrac{1}{2}\right)\right) \quad , \quad j = 0, 1, 2, \ldots \quad . \tag{2-26}$$

Table 2-3 lists the pixel count per ring for the eight possible choices of n_0 ; shaded rows are undesirable by reason of previously discussed criteria for aspect ratio and asymmetry.

n_0	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_{10}
1	7	13	19	26	32	38	44	51	57	63
2	8	14	21	27	33	40	46	52	58	65
3	9	16	22	28	34	41	47	53	60	66
4	10	17	23	29	35	42	48	54	60	67
5	11	17	24	30	36	42	49	55	61	68
6	12	18	24	31	37	43	50	56	62	68
7	13	19	25	32	38	44	50	57	63	69
8	14	19	26	32	38	45	51	57	63	70

Table 2-3. Pixel count per outwardly growing ring

There is no limit to *j*; table 2-3 stops at $j=10$ for convenience. For smooth transition to log-polar periphery, it is advisable to take advantage of n_j's which are close to powers of 2 (if pyramiding is desired) or at least even numbers. For example, note that $n_5 = 32$ and $n_{10} = 63$ for $n_0 = 1$. Thus 5-ring and 10-ring foveas yield good transitions to log-polar peripheries of $n=32$ and $n=64$, respectively. Figure 2-8 illustrates some foveas corresponding to table 2-3.

Note from equations 2-23 and 2-24 that ring area increases by 2π for each ring, i.e.,

$$A_{j+1} - A_j = 2\pi \ . \tag{2-27}$$

Thus, the number of pixels increases by about 2π per ring, i.e. usually 6 but sometimes 7 by virtue of the $2\pi - 6$ residue, analogous to leap-year day-count. Table 2-4 lists the actual pixel increment per ring, i.e. the difference between adjacent cells in table 2-3.

n_0	n_1 $- n_0$	n_2 $- n_1$	n_3 $- n_2$	n_4 $- n_3$	n_5 $- n_4$	n_6 $- n_5$	n_7 $- n_6$	n_8 $- n_7$	n_9 $-n_8$	n_{10} $- n_9$
1	6	6	6	7	6	6	6	7	6	6
2	6	6	7	6	6	7	6	6	6	7
3	6	7	6	6	6	7	6	6	7	6
4	6	7	6	6	6	7	6	6	6	7
5	6	6	7	6	6	6	7	6	6	7
6	6	6	6	7	6	6	7	6	6	6
7	6	6	6	6	7	6	6	7	6	6
8	5	6	7	6	6	7	6	6	6	7

Table 2-4. Pixel increment per outwardly growing ring

Figure 2-8 illustrates some of the foveas specified by table 2-3.

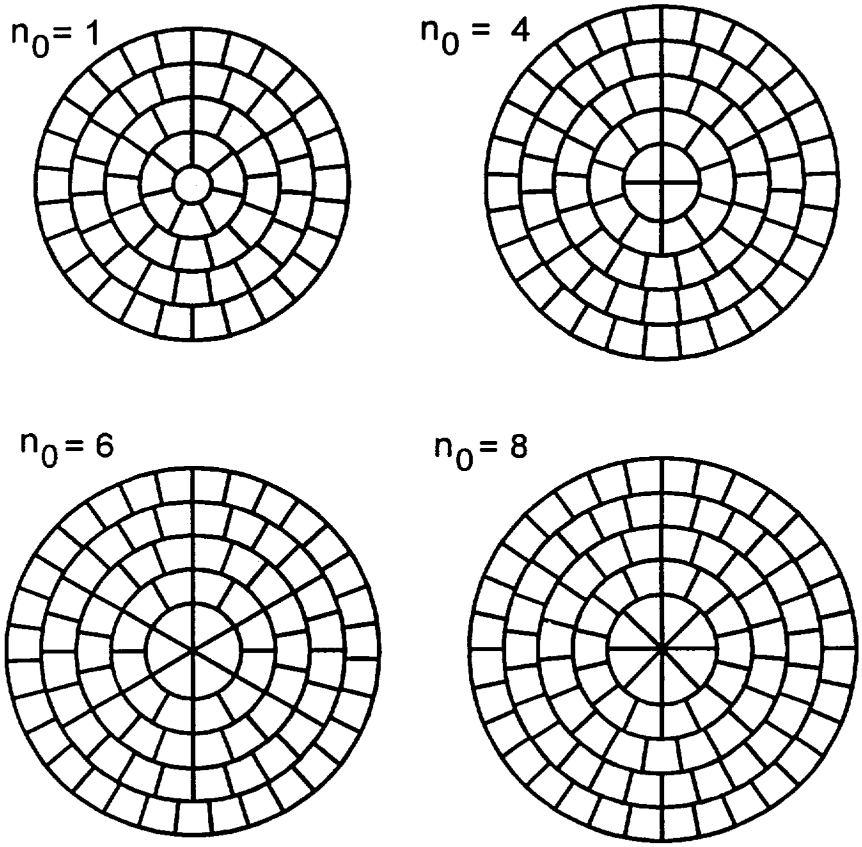


Figure 2-8. Outwardly grown foveas

Figure 2-9 plots area per pixel in successive rings for the eight choices of n_0 . Note that worst case errors of less than 5% (from the desired unit area) occur in the inner rings of some configurations, but in all configurations errors diminish to 1% and less after about five rings where their quantization errors are diluted amongst higher pixel counts.

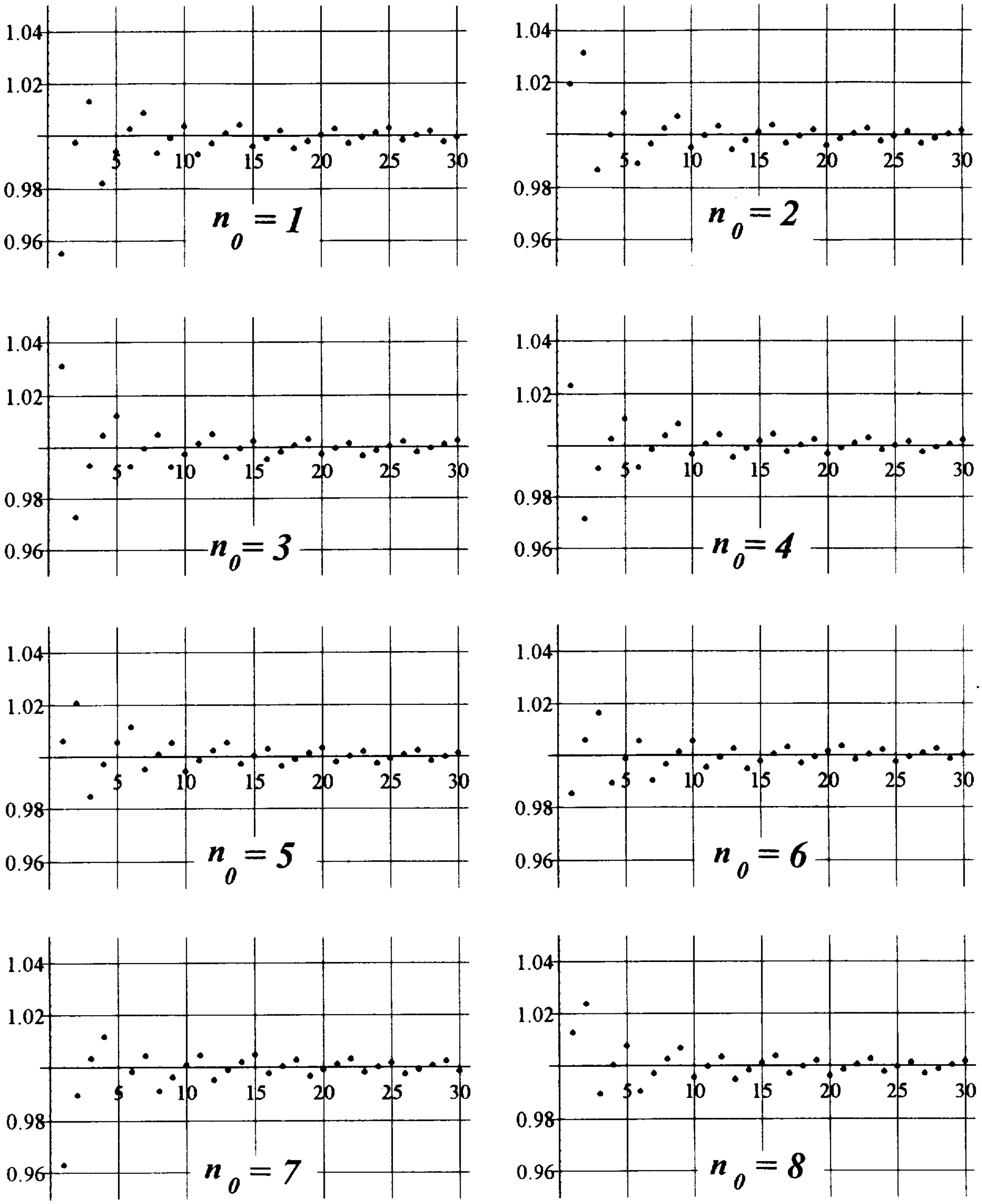


Figure 2-9. Pixel area for outwardly grown foveas

Yet another strategy for building foveas outward from the center comes from the observation that there is a predominance of 6's in table 2-4. By *forcing* pixel increment to 6 per ring, we can generate more regular foveas whose pixel counts are not subject to the irrational caprices of quantizing multiples of 2π. In order to avoid systematic area error build up, the trick is to assure that ring radius is proportional to ring count. For unit ring thickness, this requires that ring midlines (figure 2-5) fall on integer values, and hence ring boundaries half way in between at

$$r_j = j + \frac{1}{2} \quad ; \; j = 0, 1, 2, 3, \ldots \tag{2-28}$$

Then pixel count is

$$n_j = 6\,j \; ; \; j = 1, 2, 3 \ldots \tag{2-29}$$

The midline arc length of a pixel (and also its area, because ring thickness is unity) is thus

$$m_6 = \frac{2\pi\, j}{6\,j} = \frac{\pi}{3} \approx 1.047 \tag{2-30}$$

yielding an aspect ratio within 5% of unity while maintaining unit pixel area throughout the fovea.

Some multiples other than six, namely four and eight, yield more attractive foveal symmetries which blend with power-of-two log-polar peripheries, at the cost of somewhat less isotropic aspect ratios. That is,

$$n_j = 4\,j \; ; \; j = 1, 2, 3 \ldots \tag{2-31}$$

and

$$n_j = 8\,j \; ; \; j = 1, 2, 3 \ldots \tag{2-32}$$

yield midline arc lengths of

$$m_4 = \frac{2\pi\, j}{4\,j} = \frac{\pi}{2} \approx 1.57 \tag{2-33}$$

and

$$m_8 = \frac{2\pi\, j}{8\,j} = \frac{\pi}{4} \approx 0.78 \quad . \tag{2-34}$$

The respective aspect ratios are the reciprocals, .64 for m_4 and 1.28 for m_8. Since every power of two beyond eight is a multiple of four and eight, any power-of-two log-polar periphery can blend perfectly by using the outer ring of the four- or eight-multiple fovea as its inner ring. Power-of-two pixel counts are desirable for image pyramiding. Figure 2-10 illustrates foveas for these four-, six- and eight-multiple foveas.

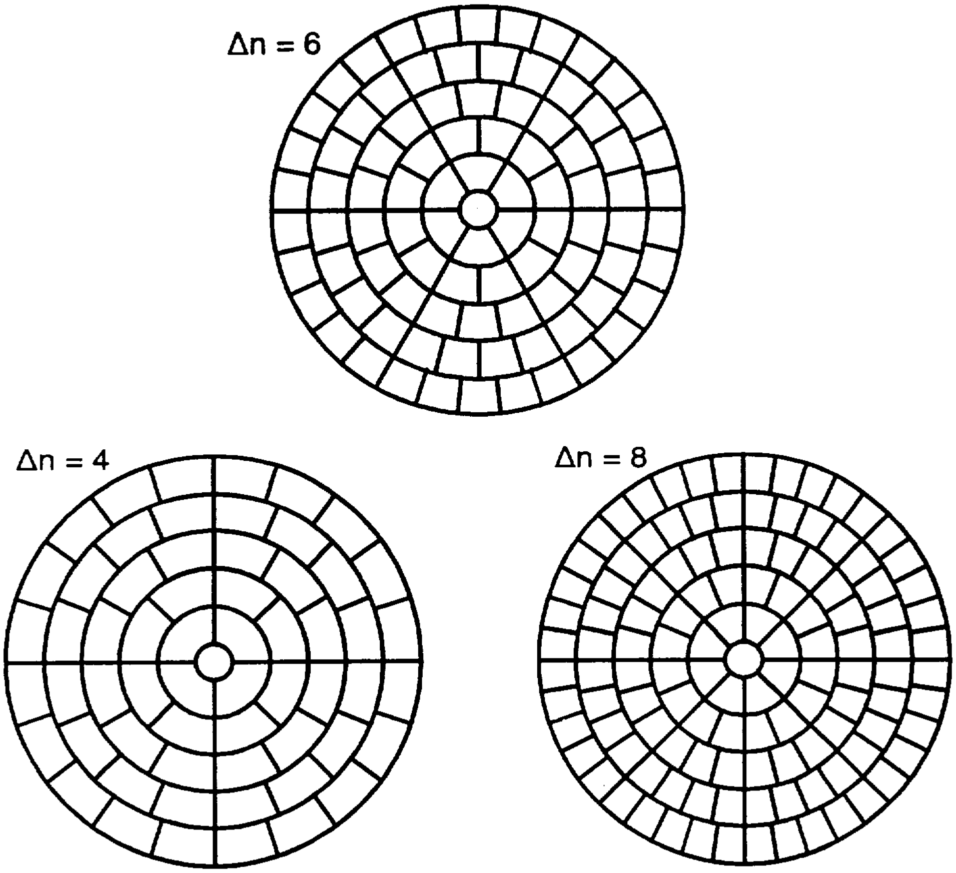


Figure 2-10. Foveas based on multiples 4, 6, and 8 per ring

2.4 Hexagonal-to-Polar Blended Foveas

The preceding sections described polar, i.e. rotationally symmetric, foveas, but traditional image processing is based on linear grids which arise naturally from row-column pixel patterns. Rectilinear (Cartesian coordinate) based techniques such as correlation, template matching, and centroid finding could be very useful in the fovea for binocular registration and pattern recognition. We now describe a method of seamlessly blending a linear-coordinate central fovea with a curvilinear log-polar periphery.

Linear coordinate tessellations of regular polygons include square, triangular, and hexagonal tilings. Of these, hexagonal tiling provides the most isotropic substrate for a variety of local image processing computations such as edge detection and associated differential measures, e.g., gradients. All nearest neighbors are the same distance from the center pixel, whereas diagonal neighbors in square coordinates (the so-called quadratic lattice) are more distant by 41%. Hexagonal tessellations also admit 3-color tilings for color vision. Hexagonal neighborhoods are the most nearly "round" of the choices listed above; therefore they can be blended to polar symmetry with the least amount of distortion.

The small solid circles in figure 2-11 depict the centers of pixels in a hexagonal (Cartesian) tessellation with unit spacing. Each successive shell (solid line perimeter) contains six more pixels than its concentric predecessor. Indefinite progression of this construction would yield a purely hexagonal grid. But, suppose we progressively warp this configuration by moving the cell centers outward towards their corresponding shaded circles (*jth* hexagonal shell to *jth* circle), with 100% warping achieved at some predetermined ring *jmax* which terminates the fovea and begins the log-polar periphery. More precisely, given any pixel center at vector position *[x,y]* in the j^{th} shell, its distance from the corresponding circle is

$$d_j([x,y]) = j - \sqrt{x^2 + y^2} \quad . \tag{2-35}$$

Thus we can nudge each point to close this gap proportionally to its distance from the center to the *jmax* radius at which the gap must be reduced to zero by applying the transformation

$$[x,y] \Rightarrow [x,y]\left(1 + \frac{j}{j\max} d_j([x,y])\right) \quad . \tag{2-36}$$

This blended configuration, illustrated in figure 2-12, yields the best of both worlds (rotational and translational symmetry). Pixel count increases by six per ring, just as in the polar 6-multiple characterized in equation 2-30 and illustrated in figure 2-10. Translation in the fovea along the lines of the hexagonal grid blends to a fisheye distortion at the periphery. Rotation of imagery throughout the field of view is effected by shift proportional to ring number in the fovea, and by a constant amount in the log-polar periphery. Near the center, rotation is polygonally distorted along the lines of the hexagonal perimeters. The tradeoff between translation distortion and rotation distortion can be controlled by choosing any monotonic interpolation function rather than the linear one of equation 2-36.

The peripheral log-polar distribution in figure 2-12 is also based on hexagonal tessellation. The spacing is defined by superimposing a hexagonal rather than Cartesian grid on figure 2-1b, and mapping the images of centers of circlets to figure 2-1a via the conformal mapping $z = e^w$.

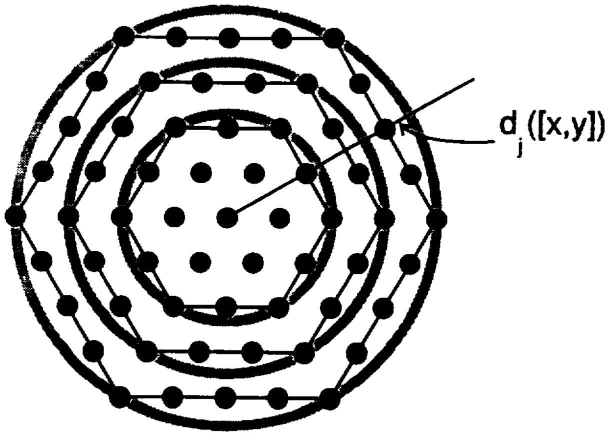


Figure 2-11. Hexagonal fovea

The price paid for blending linear foveas to polar peripheries is that the area of pixels is no longer constant in the fovea, but increases by a few percent towards the periphery. In figure 2-12 we have evaded the problem of drawing warped pixel boundaries by depicting pixels as circlets of constant radius, centered at the centers of hexagonal tessellation. In the following sub-section, we describe this and other geometric constructions for defining pixel boundaries in locally irregular tessellations.

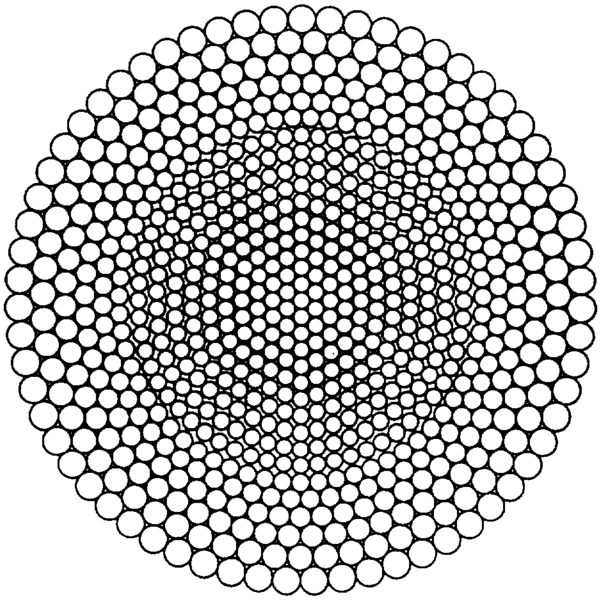


Figure 2-12. Hexagonal-to-polar blended fovea

2.5 Alternative Pixel Boundary Geometries in Biology and Silicon

The geometry of pixel boundaries is a separate issue from the geometry of pixel distribution. The latter can be characterized in terms of the distribution of the centers of pixels. In most biological examples, e.g. the compound eyes of insects or the retinas of mammals, local distribution is nearly hexagonal. This doubtless results from close packing of nearly identical units with simple perimeters. In most of the preceding examples, the local pattern of pixel center distribution is rectilinear or hexagonal. In the locally rectilinear case (which is illustrated, for example, in the log-polar periphery of figure 2-1a), the image plane is partitioned into mutually exclusive pixels whose boundaries are minimally expressed in polar coordinates, i.e. constant *r* or constant θ. This is convenient for computer graphics programming, but other boundary strategies may be more appropriate for fabricating photosensors. These include round pixels such as shown in figure 2-12 (whose interstices are gaps in the image plane), Voronoi tessellation (no gaps), and general tilings of the plane, mapped to log-polar coordinates. Figure 2-13 illustrates some Voronoi tessellations based on 1-fold and 4-fold central segmentation. Voronoi tessellation provides the most efficient packing, but could present layout problems using current VLSI fabrication techniques.

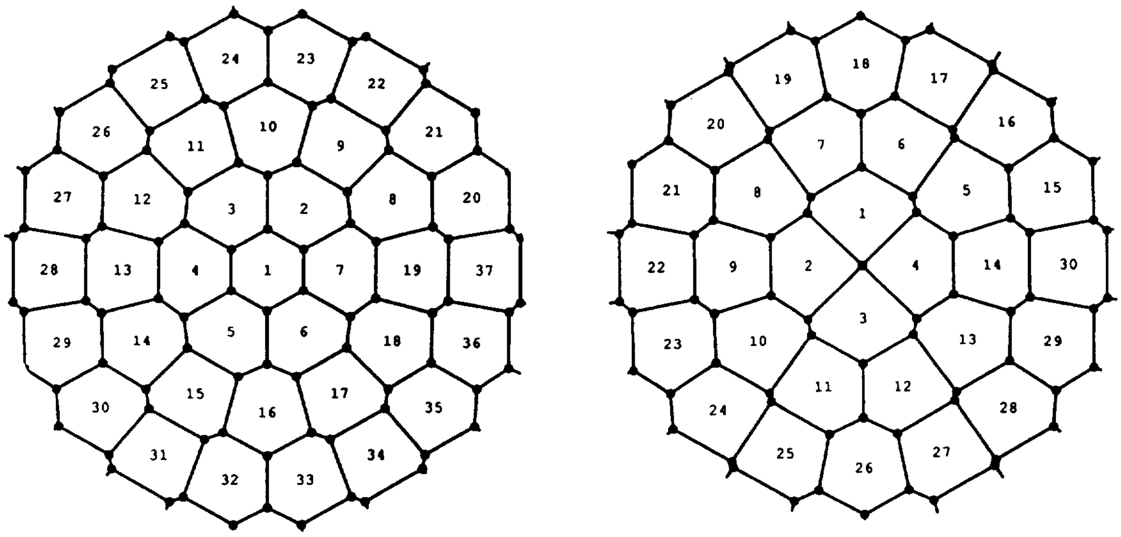


Figure 2-13. Voronoi pixel boundaries

In fabricating artificial retinas of irregular local geometry, one solution is to lay down photocells which occupy a small area at the center of the pixel, and use microlenses to focus light to them. Thus light-gathering efficiency is preserved, perimeter geometry can be ignored, and there is space for local neighborhood circuitry for "silicon retina" processing.

We conclude with a qualitative comparison of the human fovea and an artificial fovea with similar parameters, designed using the rules set forth in the preceding paragraphs. Figure 2-14 is a drawing of the human fovea by the microanatomist [Schultze, 1866] who discovered and named the rods ands cones of the human retina. Figure 2-15 is an artificial design based on the 6-multiple outwardly grown fovea described in section 2.4 illustrated in figure 2-10. Pixels are represented by circlets to mimic the cone geometry of the biological example. The perimeter blends to a log-polar distribution with $n=198$ pixels per ring. The fovea region corresponds to the central 1/2 ° of the human visual field.

- - - - - - - - - - - - -

In summary, we have shown a number of constructions which fill the central hole of a log-polar retina with a fovea consisting of pixels of uniform highest resolution, blending seamlessly at the boundary. ***In the next section we show how such sensor arrangements project a high resolution steerable "spotlight" into the center of a wide field of view.***

Primate visual anatomy and behavior exhibit just such an intimate interplay between graded resolution sensors and active vision. The speed and efficiency of such systems can be highly beneficial to the performance of robotic vision systems.

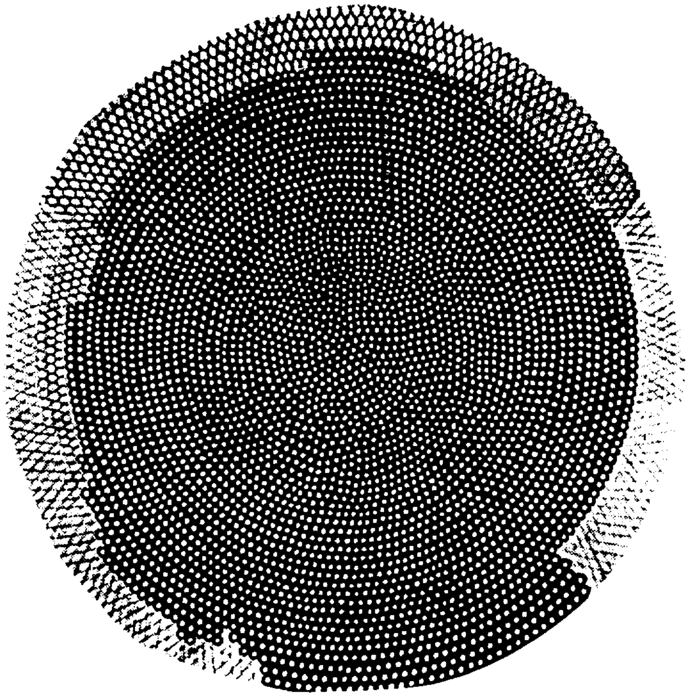


Figure 2-14. Schultze's drawing of the human fovea

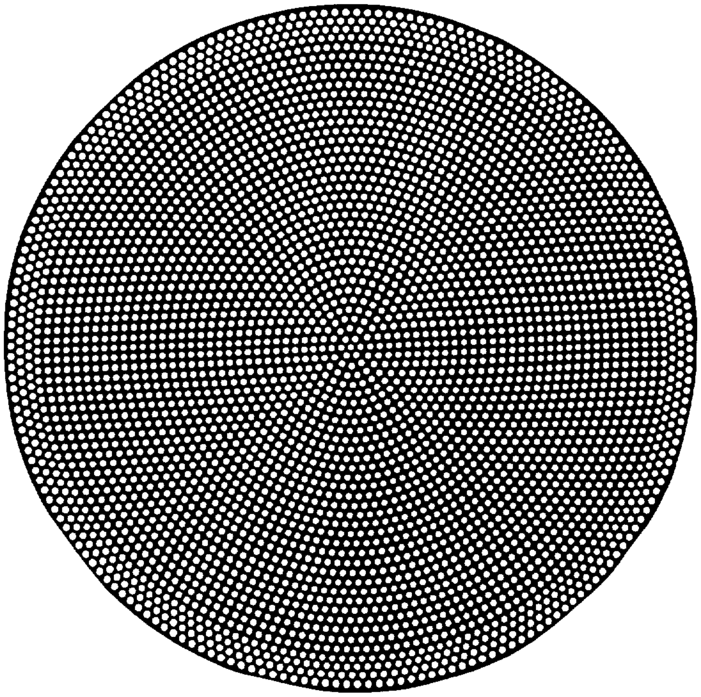


Figure 2-15. Artificial fovea and surrounding periphery matching human parameters

3.0 BINOCULAR STEREO THROUGH LOG-POLAR RETINAS

Binocular stereo vision appears at first sight to be a tractable subset of the general robot vision problem because of its deterministic geometry: triangulation of points in space as seen in two images taken from different viewpoints. The main difficulty encountered in practice is the so-called *correspondence problem*, namely, identifying the same point in two images. Active vision proposes to simplify the correspondence problem by tracking a target with two cameras once correspondence has been established, by whatever initial means. Thus, the correspondence problem does not have to be solved over and over again in real time; tracking maintains correspondence. ***Log-polar image coordinates both benefit the binocular tracking process and require it for efficient perception.*** The tracking process benefits from the low pixel count, the high precision of foveal tracking, and the wide field of view for peripheral attention. Conversely, log-polar coordinates require active vision to steer the high resolution central field of view to points of interest.

Two cameras in general position define two coordinates frames, each specifiable by six degrees of freedom with respect to environment coordinates. Stereo depth inference inverts the image projection transformations through these coordinate frames, based on intersection of the viewrays to the target point as shown in figure 3-1.

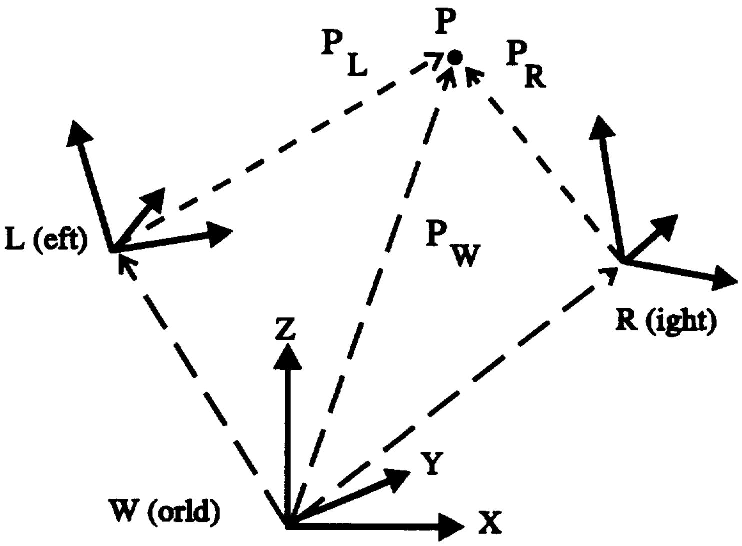


Figure 3-1. World and camera coordinate systems

Such a general formulation yields little insight into symmetries relevant to binocular stereo. Control, mechanics, and comprehension are all improved by constraining degrees of freedom, as described in the design below.

3.1 Binocular Camera Configuration with 4 Degrees-of-Freedom

For this project, we designed a high performance servo controlled camera platform which emulates much of the performance of the human eye-head system, in a simplified four degree-of-freedom configuration, illustrated in figure 3-2. Whereas the human eyeball exhibits three rotational degrees of freedom, each camera in our system has only one. Whereas the human neck exhibits three more degrees of freedom, we mount the vergence system on a pan-tilt mechanism with two degrees of freedom. These simplifications reduce the complexity of mechanics and controls while allowing gaze in arbitrary directions with arbitrary choice of vergence angle within the limits of excursion. Details of mechanism and control systems are given in sections 5 and 6. Binocular coordinate geometry is given below.

Figure 3-2 illustrates two cameras separated by baseline B with principal axes A_L (left) and A_R (right). These are mounted on servo-controlled motors with parallel vergence axes V_L and V_R perpendicular to A_L and A_R through the camera nodal points (centers of perspective projection). The motors are in turn attached to a horizontal crossbar which tilts about a horizontal axis H through the camera nodal points. Thus vergence and tilt are pure rotations about the camera nodal points. The central pan axis, V_C, is the vertical bisector of the H axis.

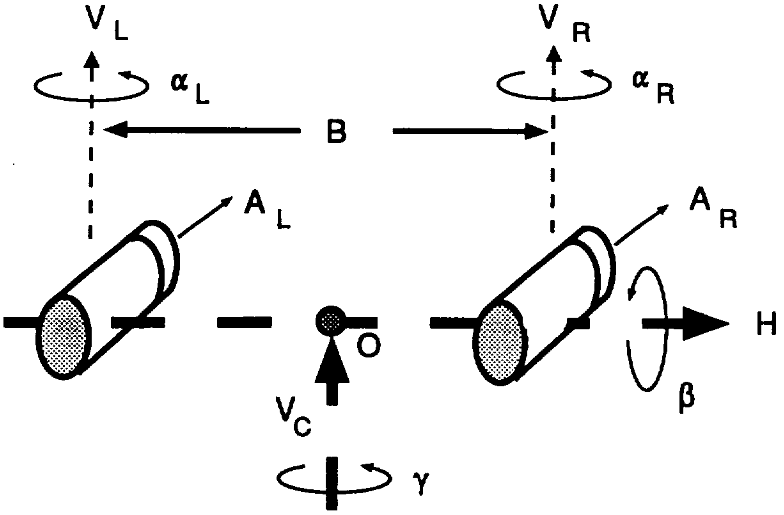


Figure 3-2. Articulation of camera mount

The general stereo triangulation problem, i.e. locating a point in space from its images in the two cameras, is solved by inverting coordinate transformations and perspective projection as follows. Consider the 3-D coordinates of a point P in space with respect to left (L) and right (R) camera coordinate frames as shown in figure 3-1,

$$P_L = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_L \quad ; \quad P_R = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_R \tag{3-1}$$

where the X-axis is horizontal and Y-axis vertical (negative V_L and V_R in figure 3-2), both parallel to the image plane, and Z-axis outward perpendicular to the image plane (positive A_L and A_R in figure 3-2).

Transforming these coordinates to the center of the H-axis, by undoing vergence angles α about the Y-axes and offset $B/2$ from the "Cyclopean" [Julesz, 1971] center ("O" in figure 3-2), yields "head" coordinates

$$P_H^L = \begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix}_H = \begin{bmatrix} \cos\alpha_L & 0 & -\sin\alpha_L \\ 0 & 1 & 0 \\ \sin\alpha_L & 0 & \cos\alpha_L \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_L - \begin{bmatrix} B/2 \\ 0 \\ 0 \end{bmatrix} \tag{3-2a}$$

and

$$P_H^R = \begin{bmatrix} X_R \\ Y_R \\ Z_R \end{bmatrix}_H = \begin{bmatrix} \cos\alpha_R & 0 & -\sin\alpha_R \\ 0 & 1 & 0 \\ \sin\alpha_R & 0 & \cos\alpha_R \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_R + \begin{bmatrix} B/2 \\ 0 \\ 0 \end{bmatrix} \tag{3-2b}$$

generically abbreviated as

$$P_H^M = {}^H T_M \, P_M \tag{3-2c}$$

where M is R or L and ${}^H T_M$ is the transformation which takes camera coordinates to Cyclopean coordinates.

Finally, inverting head pan γ about the vertical axis $-V_C$ and tilt (β) about the H-axis yields world coordinates,

$$\begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix}_W = \begin{bmatrix} \cos\gamma & -\sin\beta\sin\gamma & -\cos\beta\sin\gamma \\ 0 & \cos\beta & -\sin\beta \\ \sin\gamma & -\sin\beta\cos\gamma & \cos\beta \end{bmatrix} \begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix}_H \tag{3-3a}$$

and

$$\begin{bmatrix} X_R \\ Y_R \\ Z_R \end{bmatrix}_W = \begin{bmatrix} \cos\gamma & -\sin\beta\sin\gamma & -\cos\beta\sin\gamma \\ 0 & \cos\beta & -\sin\beta \\ \sin\gamma & -\sin\beta\cos\gamma & \cos\beta \end{bmatrix} \begin{bmatrix} X_R \\ Y_R \\ Z_R \end{bmatrix}_H \tag{3-3b}$$

generically abbreviated as

$$P_W^M = {}^W T_H \, P_H^M \tag{3-3c}$$

where ${}^W T_H$ is the pan-tilt transformation, W denotes world coordinates and M is L(eft) or R(ight) as before.

Now, stereo triangulation involves identifying a corresponding point in each image, projecting its viewray into world coordinates, and solving the simultaneous equations corresponding to the 3-D intersection of the two viewrays. Figure 3-3 illustrates the perspective projection of a viewray onto the image plane, which is parallel to the X-Y plane of the camera frame, one unit out (focal length units) on the Z-axis. That is, the image plane is located at $[X, Y, 1]^T$ (where superscript ^{T} indicates transpose, i.e. the column vector in accordance with the convention of our equations 3-1,2,3) and we use coordinates $[x, y]^T$ to distinguish these image plane points from those of the 3-D environment.

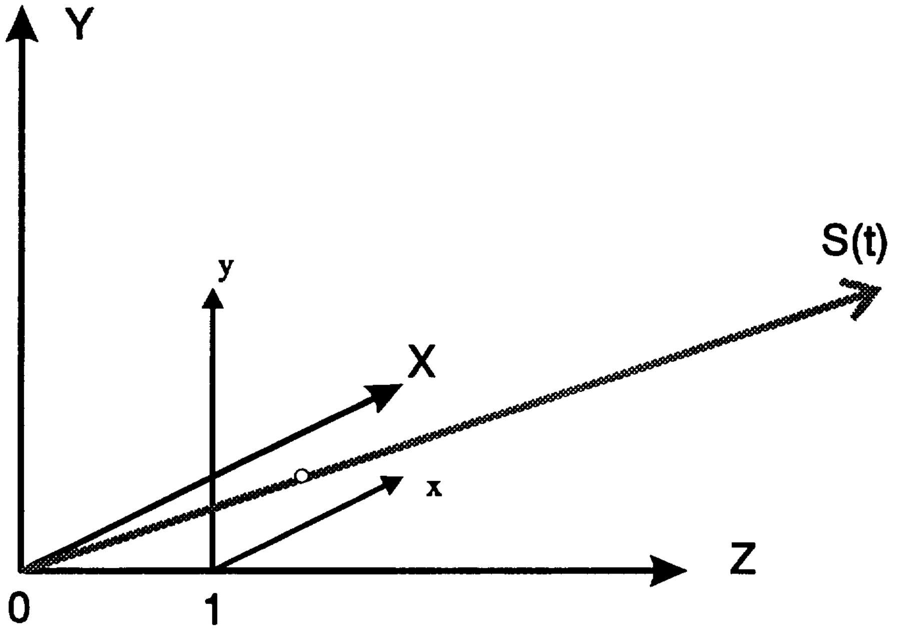


Figure 3-3. Perspective projection

Thus, by similar triangles, image plane coordinates $[x, y]^T$ are

$$x = X / Z \;\; ; \;\; y = Y / Z \;\; . \tag{3-4}$$

The inverse problem, generating the viewray, extends the image point $[x, y]_M^T$ to the ray

$$S_M(t) = \begin{bmatrix} tx \\ ty \\ t \end{bmatrix}_M \tag{3-5}$$

in camera coordinates, whence transformation to world coordinates (equation 3-3) yields equations

$$S_W^L(t_L) = {}^W T_H \; {}^H T_L S_L(t_L) \;\; ; \;\; S_W^R(t_R) = {}^W T_H \; {}^H T_R S_R(t_R) \tag{3-6}$$

which can be solved simultaneously for t_L and t_R to find the 3-D point of intersection.

The *disparity* associated with the point of intersection of these rays is the difference in positions of the images in image coordinates. That is, if the two image planes were superimposed, the disparity would be the vector connecting the point in one image plane to the corresponding point in the other, namely,

$$\begin{bmatrix} x \\ y \end{bmatrix}_L - \begin{bmatrix} x \\ y \end{bmatrix}_R \tag{3-7}$$

The equations of intersection of these rays in world coordinates can be substantially simplified by restricting pan angle α to zero. Equation 3-7 reduces to an offset in X only whence (from equation 3-4 above)

$$x_L = \frac{X + B/2}{Z} \;\; ; \;\; x_R = \frac{X - B/2}{Z} \tag{3-8}$$

and depth Z is a simple function of horizontal disparity only,

$$Z = \frac{B}{x_L - x_R} \quad . \tag{3-9}$$

This parallel camera approach was popular in early binocular vision research because of its computational simplicity and because correspondence search along synchronized horizontal video sweep lines could be accomplished efficiently in shift register hardware. The correspondence of video lines is a reflection of the property that epipolar lines are identical in the two fields of view for parallel cameras.

One problem with parallel cameras is that disparity magnitudes are large at close ranges, requiring a long "reach" in disparity measurement mechanisms. Figure 3-4 illustrates the magnitude of disparity in pixels for 512 pixel cameras with 90° field of view separated by 25 cm (10 inches). Note that at 25 cm disparity is 256 pixels. That is, at a range equal to binocular baseline, 90° fields of view overlap by only 50%. Thus the magnitudes of all disparities at this range are half the field of view. Such large offsets eliminate practicable local neighborhood search mechanisms for correspondence.

Another problem arises for active vision. Both cameras cannot simultaneously track a close object. Thus, if cameras are parallel, one must be dominant, and the disparities will be so large that there will be extensive areas in one field of view which have no corresponding loci in the other, impeding correspondence mechanisms.

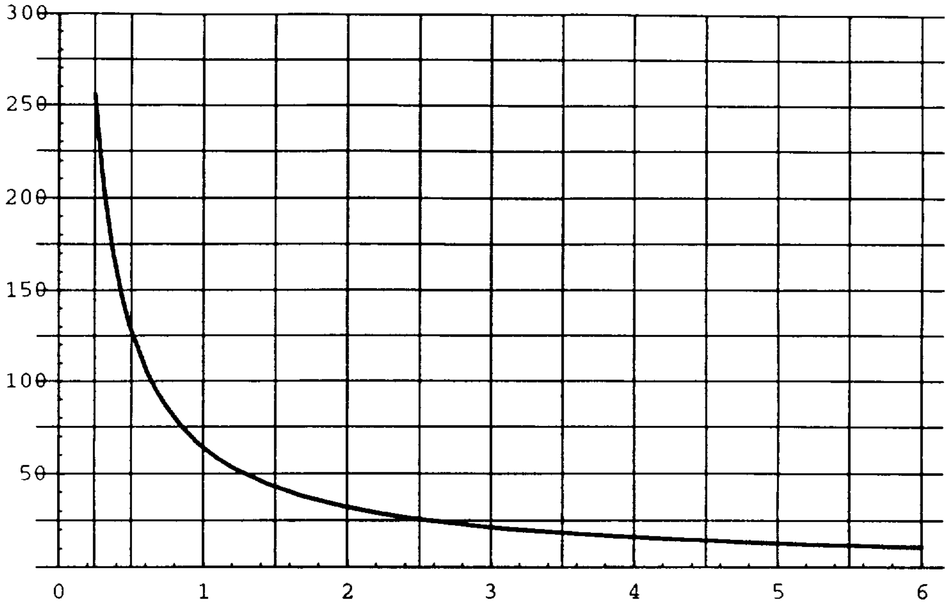


Figure 3-4. Disparity in pixels as a function of range in meters for 25 cm baseline

3.2 Intrinsic Disparity of Convergent Cameras

A natural alternative to parallel camera configuration for active vision is to converge the cameras to a target of interest so that disparity is zero at the center of the field of view [Coombs, 1992], then maintain zero disparity by tracking. This reduces the correspondence problem to a local operation and maintains the object of interest at the center of the field of view.

The following analysis is based on a camera servo algorithm which tracks a 3-D target by centering it in the fields of both cameras via vergence and servoing the head pan axis angle γ so that vergence angles are equalized, $\alpha_L = - \alpha_R$. Vergence symmetry simplifies the binocular geometry without the penalties of parallelism described above. The symmetric vergence algorithm was implemented in real time for the binocular head and is described later in section 6.

Consider figure 3-5 which is a top view of the symmetric convergent camera configuration described in the preceding paragraph. The vergence angle 2α subtended at the target P by the two camera nodal points, L and R, can be inscribed on a circle through these three points. From plane geometry (equal arcs subtend equal inscribed angles) the view rays to camera nodal points from any point P' on this circle subtend the same angle.

Thus, since the principal camera rays yield zero disparity, so do the images of all points on this circle throughout the field of view. This special circle is known as the *horopter* in visual perception. For camera imagery, objects at this locus in the environment will be in exact correspondence (same pixel address) in the two images.

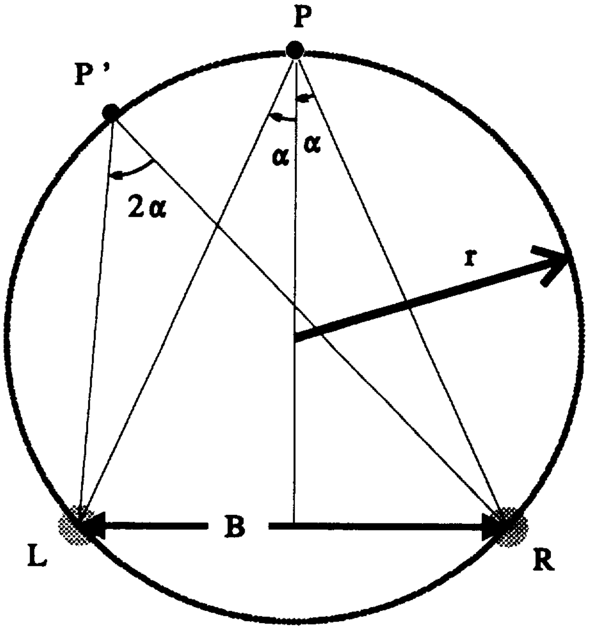


Figure 3-5. Horopter

The radius r of the horopter as a function of combined vergence angle 2α and camera baseline separation B is

$$r = \frac{B}{4}\csc(\alpha)\sec(\alpha) \tag{3-10}$$

plotted in figure 3-6 for a baseline separation B of 25 cm.

Small disparities correspond to small deviations in depth from the horopter. Such disparities can be measured by simple local neighborhood operators, to build up a dense surface map of the environment near the horopter. The Gabor filter, to be described in section 4, is an excellent operator for such purposes.

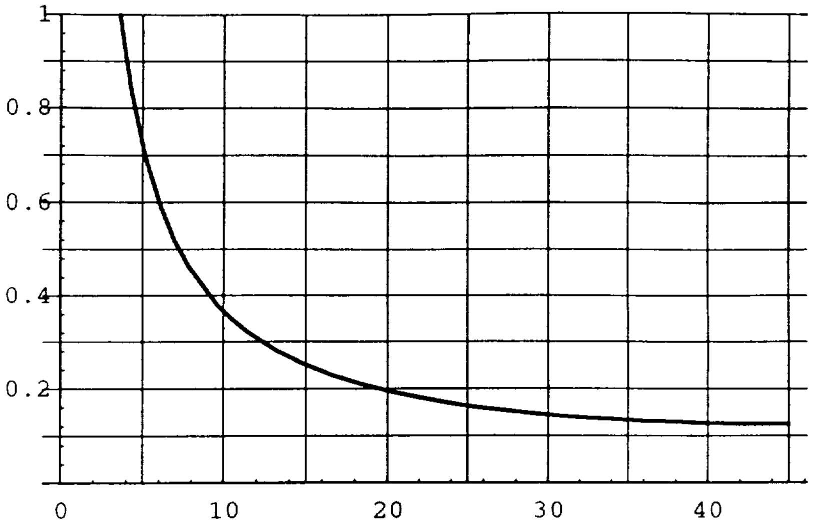


Figure 3-6. Horopter radius as a function of vergence angle (degrees)

Although zero disparity correspondence is achieved for points lying on the horopter, convergent viewing is intrinsically disparate above and below the plane containing the horopter. The nature of the problem is illustrated in figure 3-7 which depicts the superimposed left and right images of a set of lines parallel to the binocular axis, in a vertical plane tangent to the horopter at the center of the field of view, for example, horizontal lines painted on a wall directly in front of the symmetrically converging cameras.

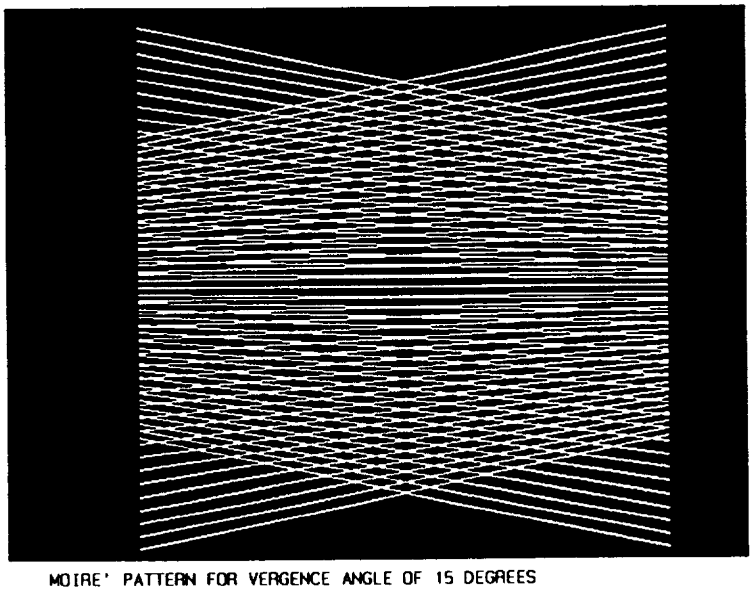


Figure 3-7. Keystone effect of convergent cameras

This perspective "keystone" effect arising from convergent camera axes introduces intrinsic disparity as follows. If viewrays are projected from identical locations $[x,y]_L^T$ and $[x,y]_R^T$ in each image plane via equations 3-5 and 3-6, they will not intersect in space, unless $x = 0$ (intersection of vertical saggital plane splitting the Cyclopean field) or $y=0$ (horopter). However, there will be some point in space on each ray where it comes closest to its counterpart. This point can be computed by taking the cross-product of the two rays to generate their common perpendicular,

$$Q(x,y)=S_W^L(1)\times S_W^R(1) \quad , \qquad (3\text{-}11)$$

finding the parallel planes perpendicular to Q containing the viewrays,

$$K_L(Q(x,y)) \quad ; \quad K_R(Q(x,y)) \qquad (3\text{-}12)$$

as shown in figure 3-8. The separation of these planes is the distance of closest approach of the corresponding "near miss" rays, $S_W^L(t_L)$ and $S_W^R(t_R)$. The points of closest approach are designated as white dots in figure 3-8.

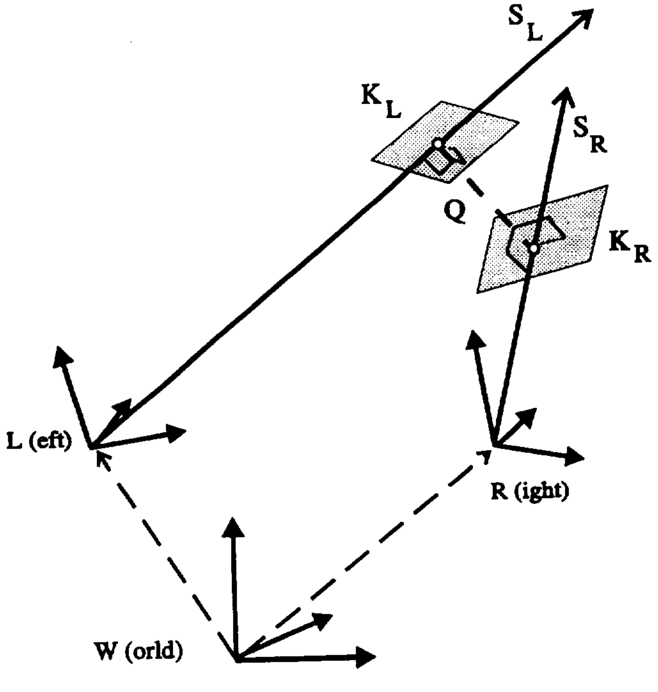


Figure 3-8. Intrinsic disparity for convergent cameras

Figure 3-9 illustrates two views of the 3-D configuration of these "near misses" by plotting the field of vectors connecting mutually closest points on corresponding viewrays. In this "pincushion" diagram, the head of the pin is on the ray from the right camera (S_R in figure 3-8) and the pinpoint terminates on the ray from the left camera (S_L in figure 3-8).

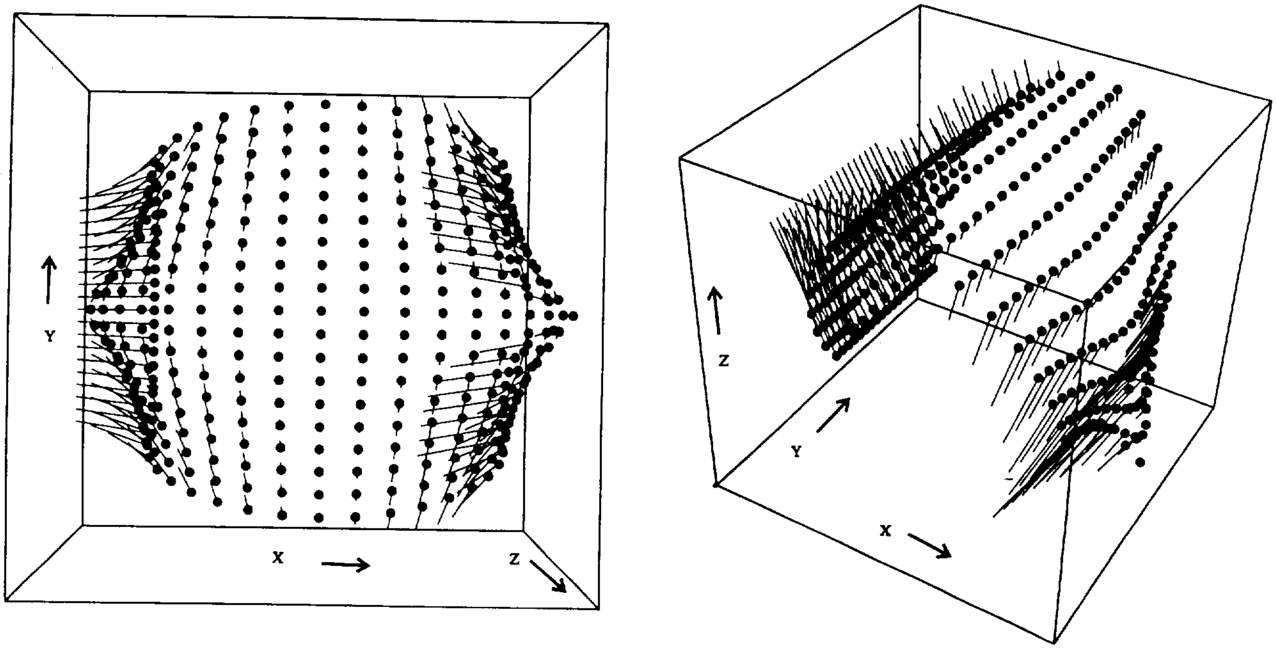


Figure 3-9. Pincushion diagram of intrinsic disparity of convergent cameras

Note that pin length is zero along the horopter and vertical line through the saggital plane, i.e., zero disparity is only possible at these loci for convergent viewing. The shape of the surface of closest approach through the midpoints of the pins is roughly cylindrical, extending the horopter in the vertical direction.

By perspectively projecting the midpoints of the "pins" in figure 3-9 to the image planes (equation 3-4) we can measure the disparity intrinsic to convergent viewing. That is, in convergent viewing, the midpoint of the pin is the 3-D locus which is most "horoptoid", i.e. generates the smallest disparity for a pair of corresponding image points. The intrinsic disparity is the difference in position of this point in both images. We omit the analytic expression here since it occupies a full page of *Mathematica*® output and lends no insight by virtue of its complexity. Instead, we plot the disparity vectors in the image plane in figure 3-10. The operational significance of the large vectors in the periphery of the four intervening quadrants is that a correspondence operator must "reach" to achieve registration in the periphery, regardless if the horopter is locked in. Note that the horizontal and vertical mid-lines display zero disparity, corresponding respectively to the horopter and saggital (vertical) plane splitting the binocular field of view.

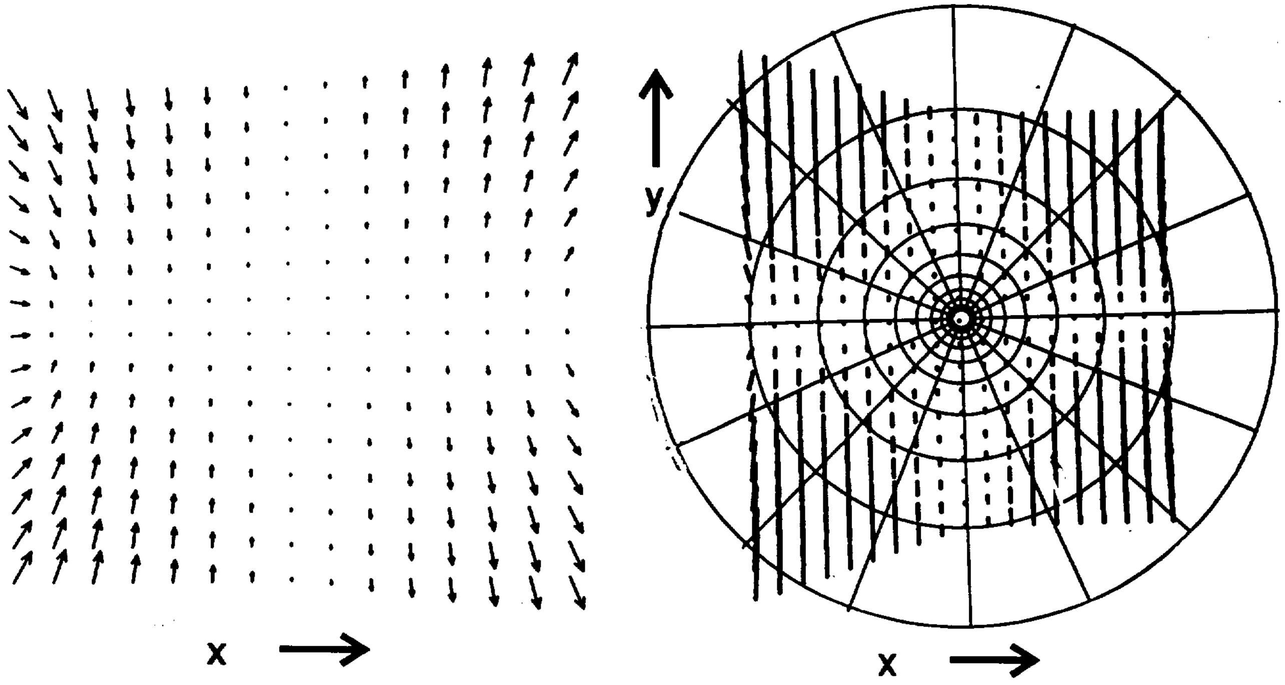


Figure 3-10. Image plane display of intrinsic disparity of convergent cameras

Figure 3-11 displays the signed magnitudes of disparity, and figure 3-12 absolute value of these magnitudes to reveal the bilateral and vertical symmetry. Figure 3-13 is a contour map of figure 3-12 to emphasize its roughly hyperboloid shape. The operational significance of the disparity magnitude functions displayed here is that to maintain binocular correspondence throughout the visual field, correspondence operators must be able to reach at least as far as the magnitude of this intrinsic disparity.

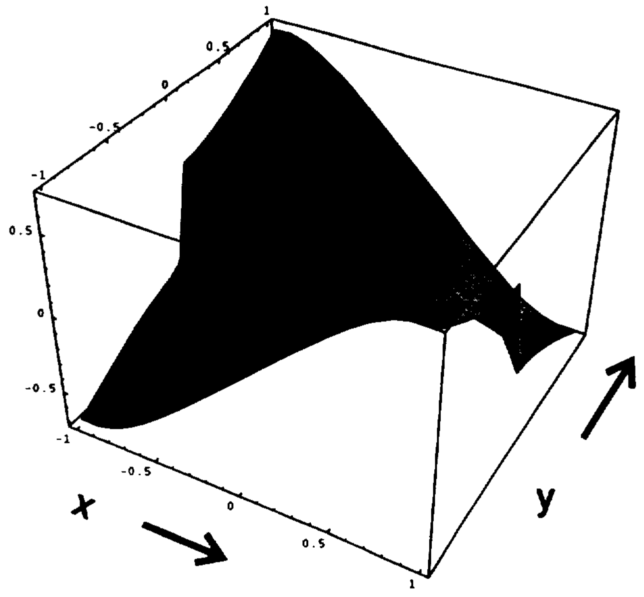


Figure 3-11. Signed magnitude of intrinsic disparity of convergent cameras

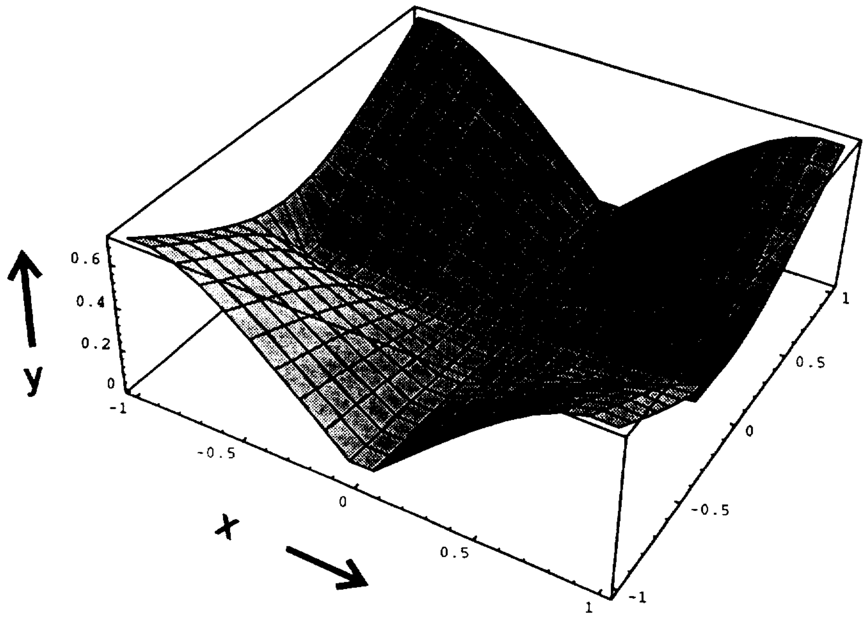


Figure 3-12. Absolute value of magnitude of intrinsic disparity of convergent cameras

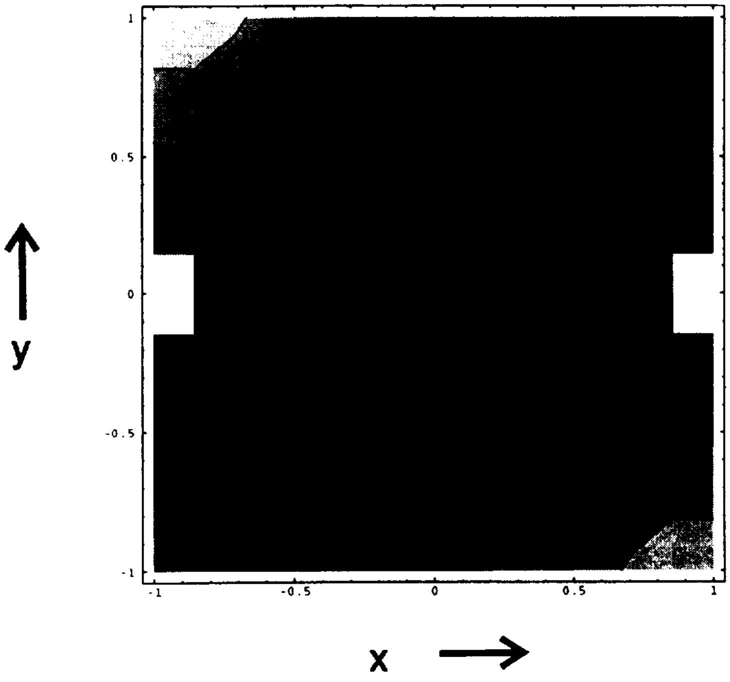


Figure 3-13. Contour map of intrinsic disparity of convergent cameras

3.3 Comparison of Disparity Fields for Log-Polar vs Cartesian Retinas

The preceding discussion described the disparity intrinsic to convergent binocular viewing, a phenomenon which is independent of choice of viewplane coordinates. That is, choosing an alternative pixel tessellation (e.g., log-polar, fisheye, etc.) would only alter the local density of points on the intrinsic disparity surface depicted in figures 3-10 to 3-13, but not its shape. The height of the surface, or the length of the vectors in figure 3-10, determines the distance which image plane operators must "reach" to achieve correspondence in that part of the field of view, regardless of pixel tessellation. An advantage of log-polar coordinates is to reduce the required reach for peripheral disparity measurements by virtue of increased pixel size, and hence increased neighborhood size, as one moves to the periphery in the corner zones separated by the horopter and the vertical plane. Figure 3-14 illustrates the signed magnitude of these disparities in log-polar coordinates displayed from two graphics viewpoints. Note that larger disparities are pushed further out (i.e. to the right in *(log r, θ)*) and that the four ridges correspond to the four quadrants of the image plane from figures 3-11 or 3-14.

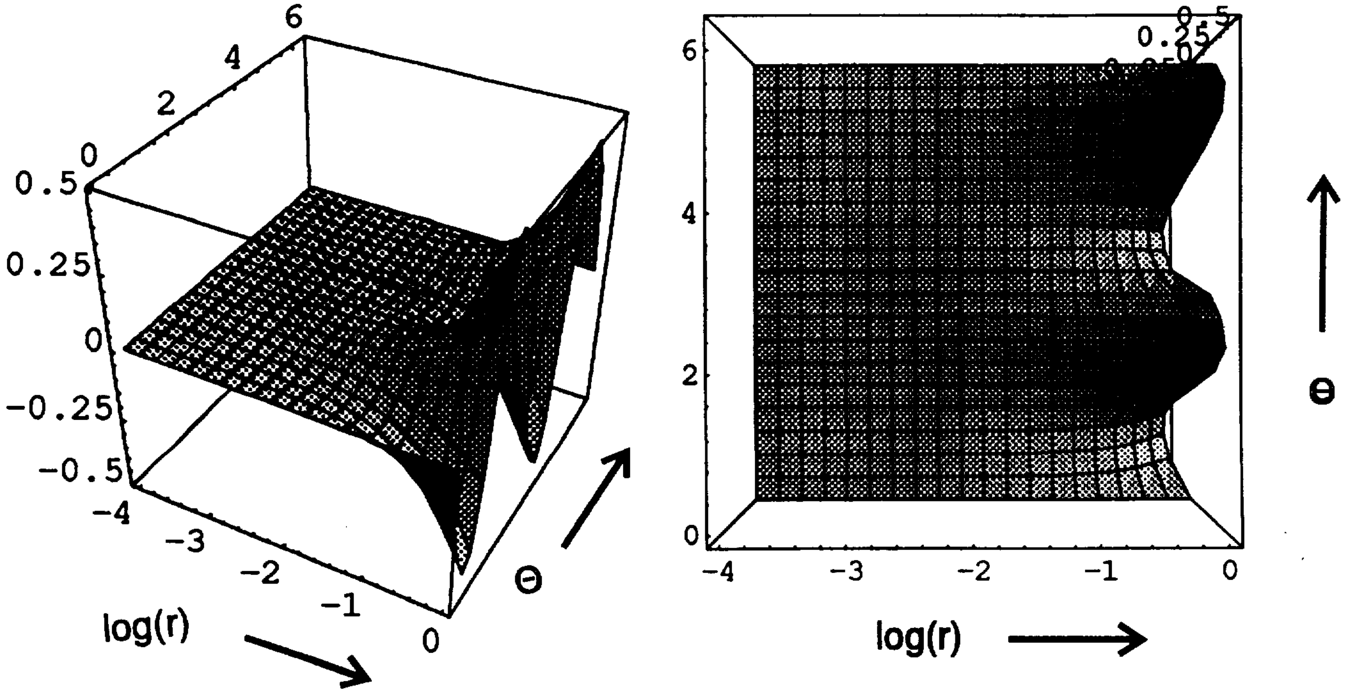


Figure 3-14 Magnitude of intrinsic disparity, mapped to log-polar coordinates

The preceding discussion described the zero-disparity curves (horopter) and near-zero disparity surface for convergent binocular viewing. Objects in the environment residing at that locus are in registration (or as nearly so as possible) in both images.

A more subtle aspect of stereo perception is the ability to measure 3-D surface structure which departs from this locus of nearly perfect registration: ridges, texture, holes, and bumps which can be characterized by small deviations from zero disparity. The ability to measure these deviations enriches vision with depth-of-field, rather than simple range-to-vergence measurements. Whereas the shape of the near-zero disparity surface is independent of image plane coordinates, measurement of such deviations is not. That is, if pixel size is non-uniform, surfaces of fixed non-zero disparity do not conform to the shape of the horopter because fixed pixel index differences do not correspond to fixed angular differences. The sizes of neighboring log-polar pixels are not equal; they are "space-variant" in the terminology of Schwartz [Rojer and Schwartz, 1990]. We examine the geometric structure of non-zero disparity below, comparing the partitioning of space by pixel projections from binocular Cartesian grids and binocular log-polar grids.

Figure 3-15 illustrates a top view of the horopter plane, dissected by the rays projecting from parallel cameras whose image planes are divided into the uniform pixels of a traditional Cartesian coordinate system. The dissection pattern is a cross-section of the volume cells (voxels) defined by the binocular intersection of the prismatic cones projecting from image plane pixels. Note that the voxels are evenly distributed in rows parallel to the image planes, a reflection of the simplistic depth-disparity property expressed in equation 3-9. That is, any fixed disparity corresponds to a single row of voxels parallel to the image planes. Successively larger disparities correspond to successively closer distances to the viewplanes. A drawback of this property is the lack of field of view overlap at close range, as described earlier.

The size of voxels indicates the uncertainty of range measurement, which grows as the square of range, i. e taking the derivative of equation 3-9,

$$\frac{dZ}{ds} = -\frac{Z^2}{B} \qquad (3\text{-}13)$$

where $s = x_L - x_R$ is disparity. Figure 3-16 illustrates the same top view of the horopter plane for convergent Cartesian cameras. Note that voxels are arranged in concentric curves about the horopter, but that the highest density (best depth resolution) is nearest the peripheral limits of the intersections of fields of view where disparities are maximum.

Figures 3-17 and 3-18 illustrate the top view of the horopter plane for cameras with log-polar image plane pixel tessellations in parallel and convergent configurations, respectively. Note that the parallel configuration is disorderly in comparison to the Cartesian case, but the log-polar convergent configuration exhibits a high resolution focus of attention at the intersection of the centers of the fields of view. That is, at the intersection of the foveal projections, voxels are smaller and characterized by more uniform aspect ratio than they are away from this zone. This voxel configuration differs markedly from that of the Cartesian case, which has no such differentiated center.

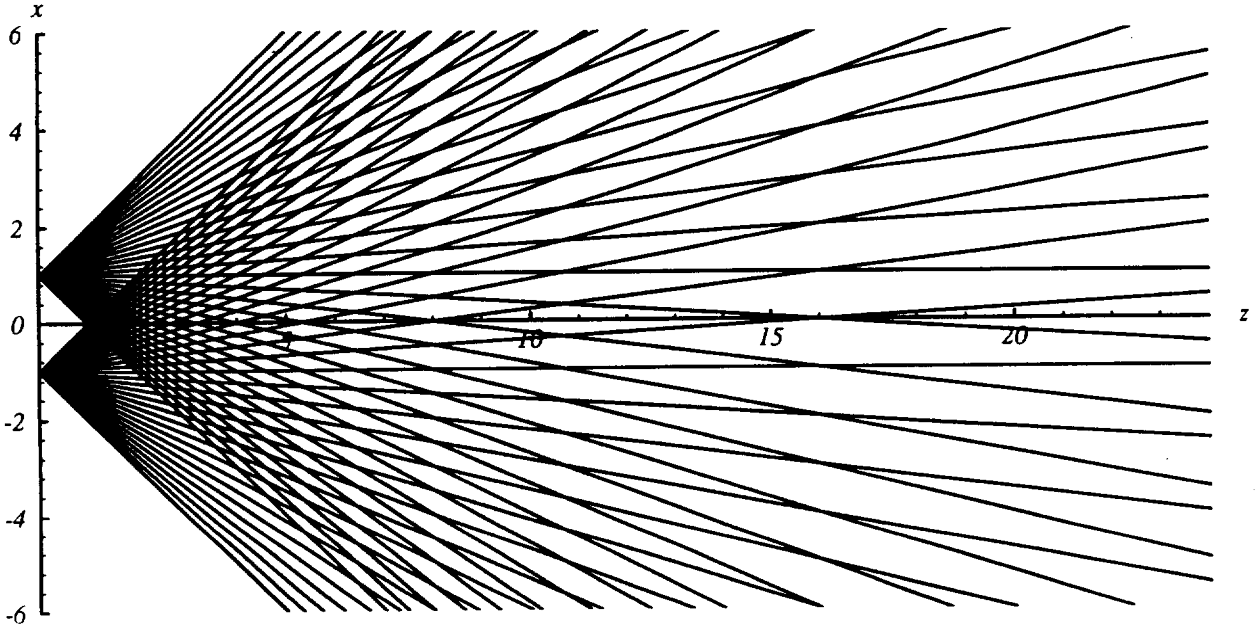


Figure 3-15. Top view of parallel Cartesian cameras

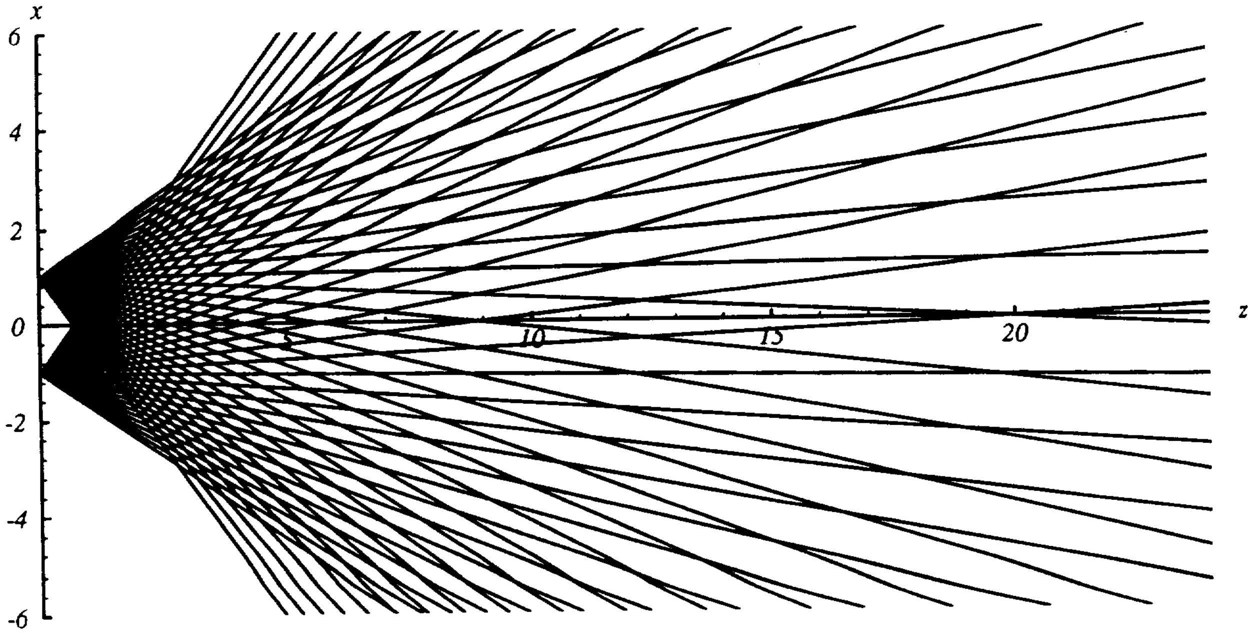


Figure 3-16. Top view of convergent Cartesian cameras

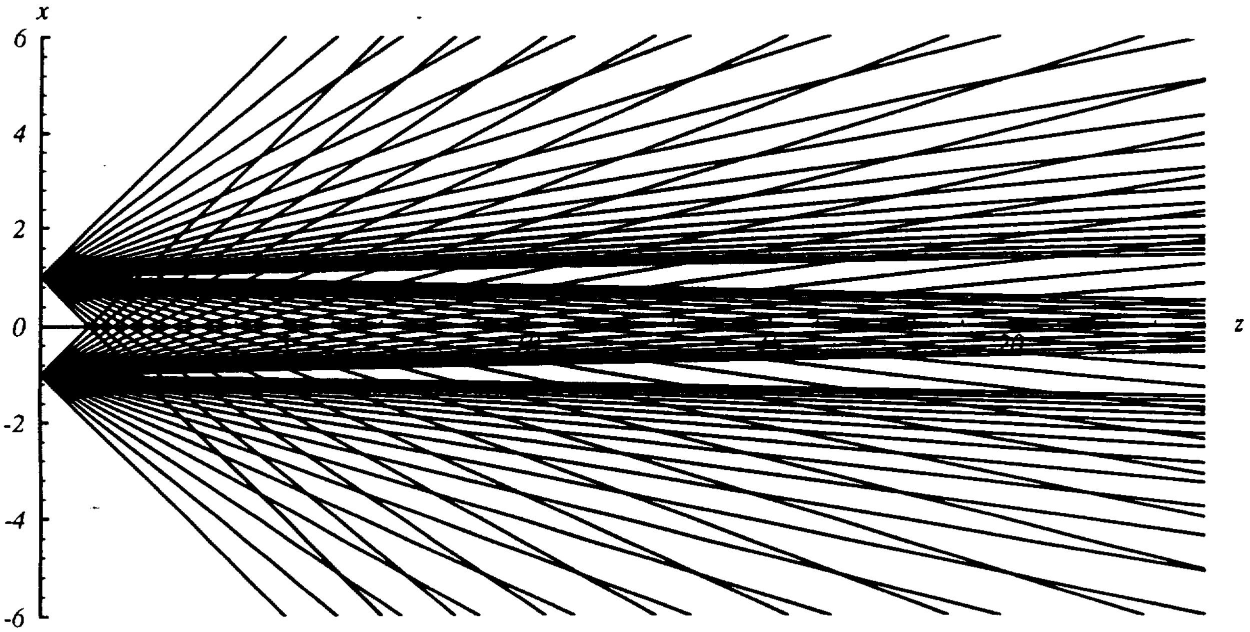


Figure 3-17. Top view of parallel log-polar cameras

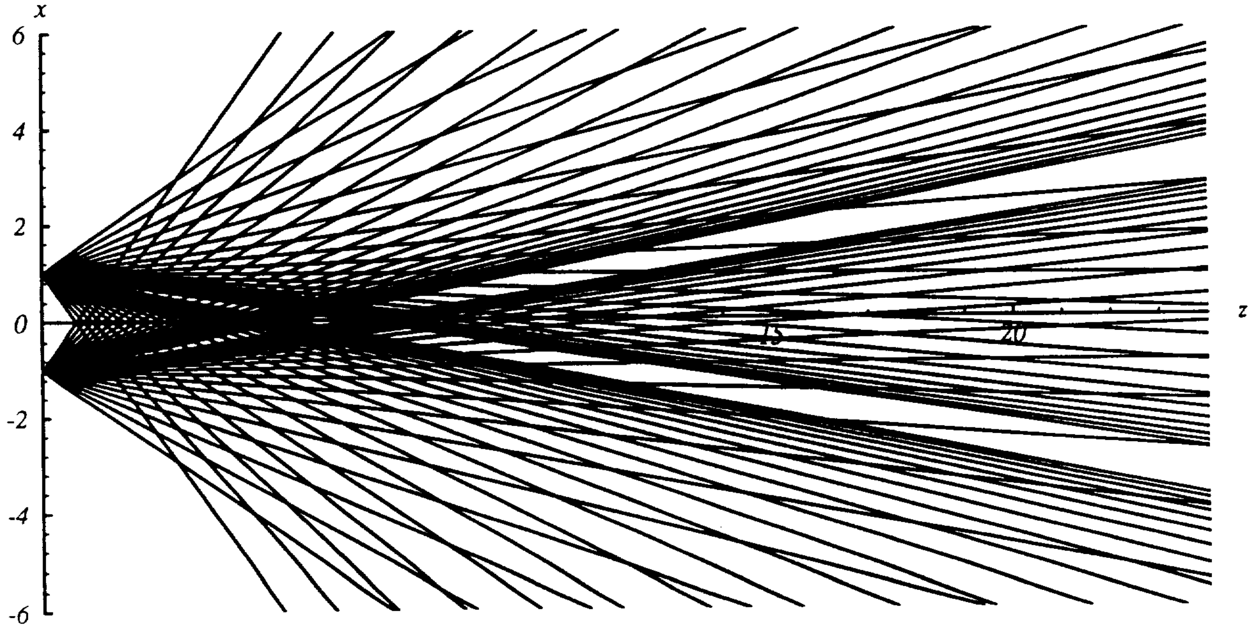


Figure 3-18. Top view of convergent log-polar cameras

We can more clearly characterize the contrasting structure of the binocular disparity fields for Cartesian and log-polar retinas by examining the iso-disparity curves which are the moiré patterns visible in figures 3-15 through 3-18. Each moiré fringe represents a constant difference in pixel index between left and right viewplane projections [Oster, 1965]. That is, indexing pixels along the horizontal image plane axis (x-axis) in the left and right fields of view as j_L and j_R respectively, a fixed disparity curve is a solution to the indicial equation

$$j_L - j_R = k \tag{3-14}$$

where k is the disparity measured in pixel count. For example, $k = 0$ corresponds to the horopter. Iso-disparity curves are found by solving the equations for the intersections of rays (e.g., equations 3-5 and 3-6) whose indices satisfy the indicial equation for fixed values of k. Figures 3-19 and 3-20 illustrate these fixed-k moiré curves corresponding to the convergent Cartesian cameras of figures 3-16 and convergent log-polar cameras of figure 3-18, respectively. The "wave fronts" (which correspond to moiré fringes) emanating from the image planes represent loci of constant disparity satisfying indicial equation 3-14. The rays which cross them correspond to loci of equal but opposite pixel index change, i.e. excursions in range resulting from changing net disparity while maintaining fixed "Cyclopean" direction of gaze. These depth-of-field rays are thus solutions of the indicial equation

$$j_L + j_R = k \ . \tag{3-15}$$

Close spacing of rays indicates fine angular (azimuth) resolution; close spacing of moiré wavefronts indicates fine range resolution. Note that in the Cartesian case (figure 3-19) the best range resolution is at the extreme periphery of the cameras' fields of view. In contrast, note that in the log-polar case, there is high resolution well within both fields of view, and that ***at the 3-D intersection of the binocular fovea projections, range-disparity voxels are nearly uniform and four times denser than the Cartesian case,*** as a result of the higher resolution allocated to the fovea. In general, the improvement in range resolution is proportional to the increase in resolution of fovea pixels over uniform Cartesian pixels. This qualitative observation will be expressed quantitatively in the next section. First, we briefly examine the vertical cross sections of voxel disparity fields to complete the picture.

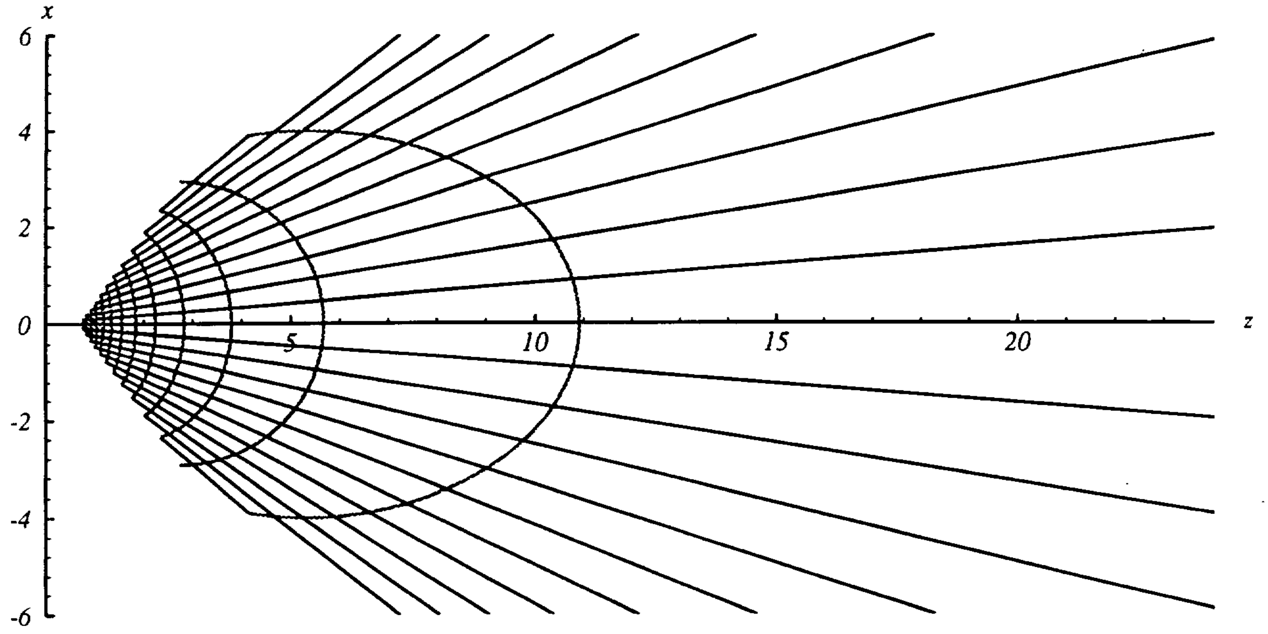


Figure 3-19. Moiré patterns for convergent Cartesian cameras

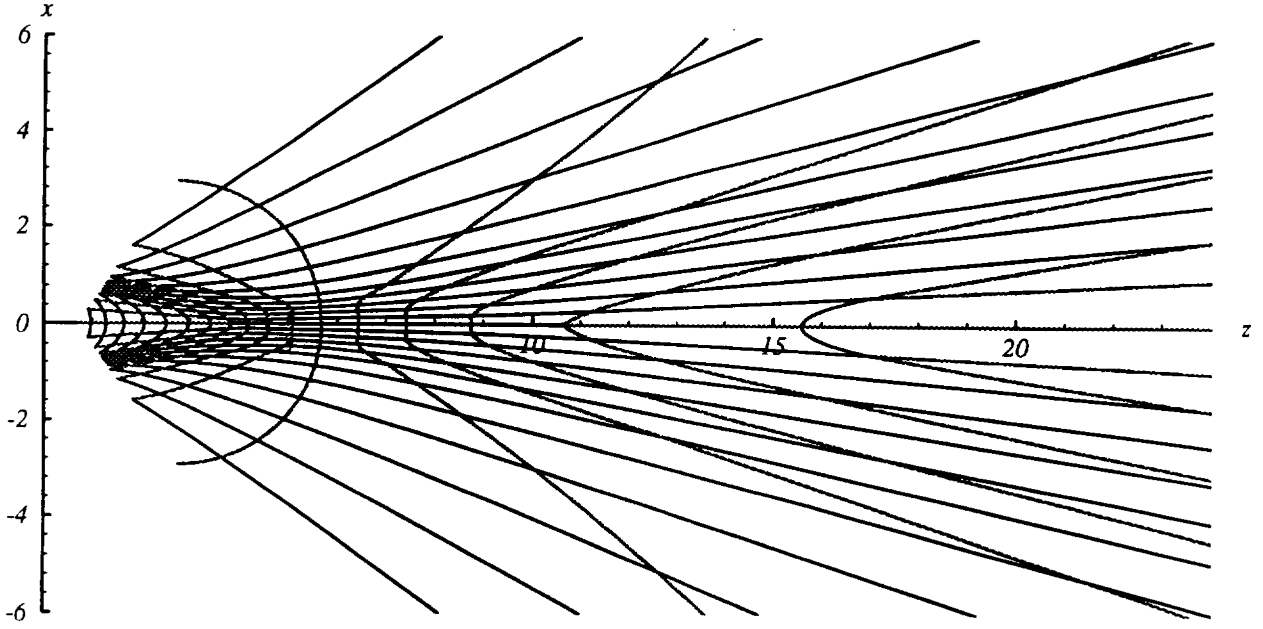


Figure 3-20. Moiré patterns for convergent log-polar cameras

Having examined the horizontal cross-section of the binocular field for symmetrically converging cameras, consider now the vertical (so-called *saggital plane* in biological vision) longitudinal section which splits the Cyclopean field of view. Figure 3-21 illustrates the intersections of pixel projections for parallel Cartesian cameras in this plane. The pencil of lines emanating from the Cyclopean origin is a cross-section of the two pencils of planes through horizontal pixel boundaries. The vertical lines are intersections of the two vertical pencils of planes projecting from camera focal points through vertical pixel boundaries. Figure 3-22 illustrates the saggital cross-section for parallel log-polar cameras. Pixel ring boundaries project to 3-D cones whose intersections with the saggital plane generate conic sections (hyperbolae of various eccentricities). Pixel ray boundaries project to pencils of planes emanating from the two principal optic axes of the two cameras, which intersect the saggital plane in horizontal lines spaced by the tangent function of equal angular intervals ($2\pi/n$ from figure 3-28b). This can be considered as an off-center longitudinal section of the projected image of the retina which is illustrated in figure 3-23. Figure 3-24 illustrates the intersection of two such cones (figure 3-23) in convergent binocular viewing, whose longitudinal section is shown in figure 3-26.

Figures 3-25 and 3-26 illustrate the saggital plane sections of the voxel field for convergent viewing through Cartesian and log-polar retinas, respectively. In the Cartesian case, note that once again the highest density (best range resolution) is at the periphery of the fields of view, closest to the cameras. The limit line of verticals has moved to the left somewhat relative to the parallel viewing configuration (figure 3-21), reflecting the introduction of vanishing points into the fields of view. In marked contrast, the log-polar convergent viewing case shown in figure 3-26 exhibits a well defined high-resolution center at the intersection of the principal viewrays. ***This focus of attention can be steered by active camera controls through the 3-D environment like the intersection of spotlights targeting any selected range and direction.***

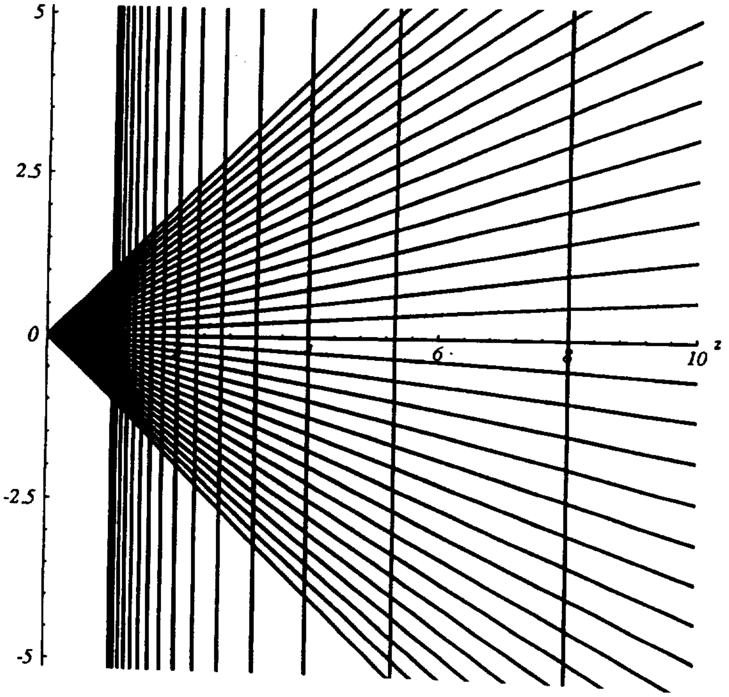


Figure 3-21. Saggital plane section for parallel Cartesian cameras

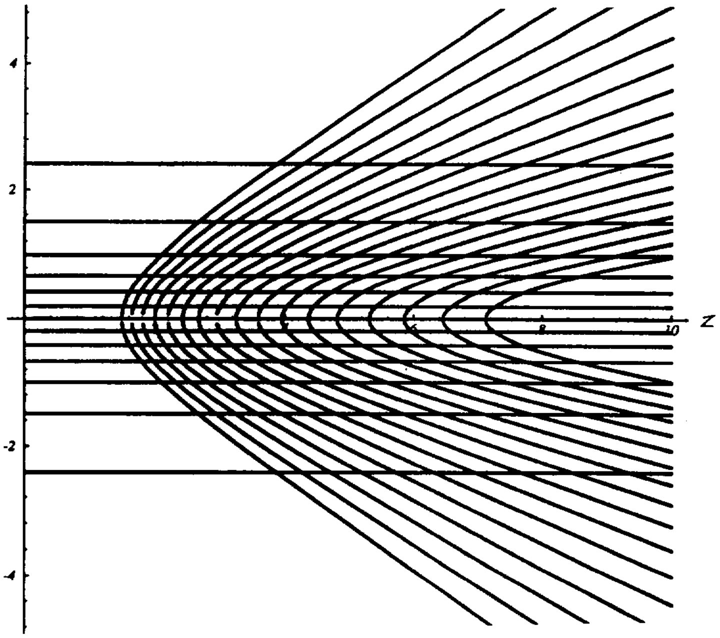


Figure 3-22. Saggital plane section for parallel log-polar cameras

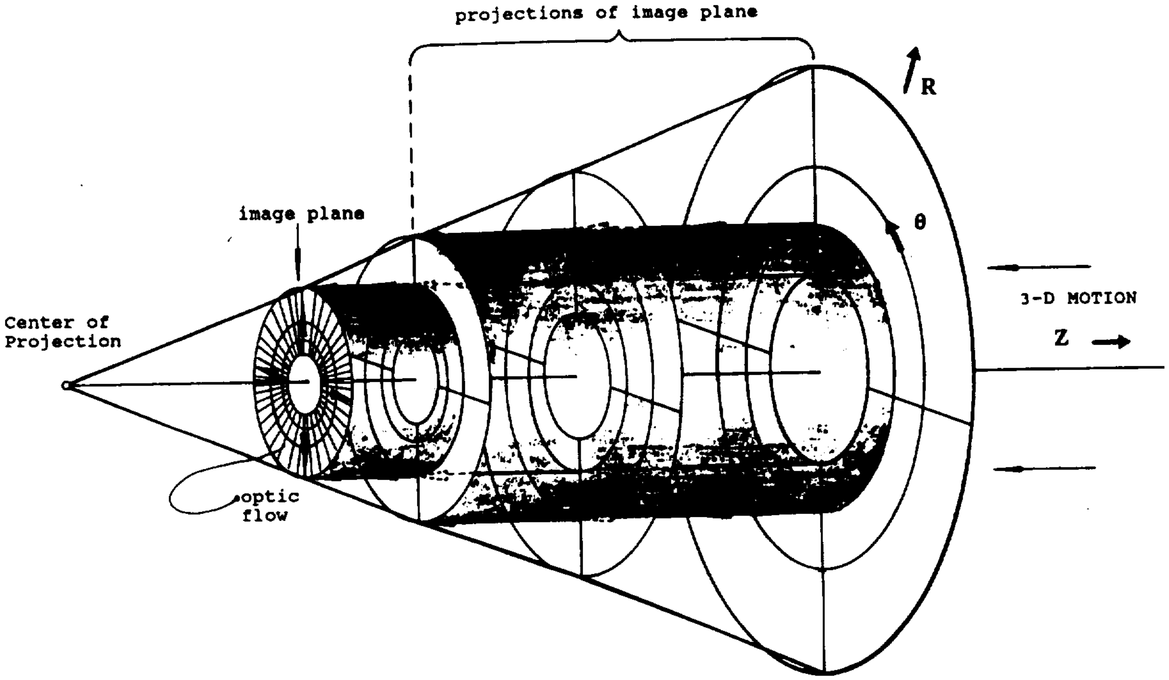


Figure 3-23. Projection of log-polar retina into visual field

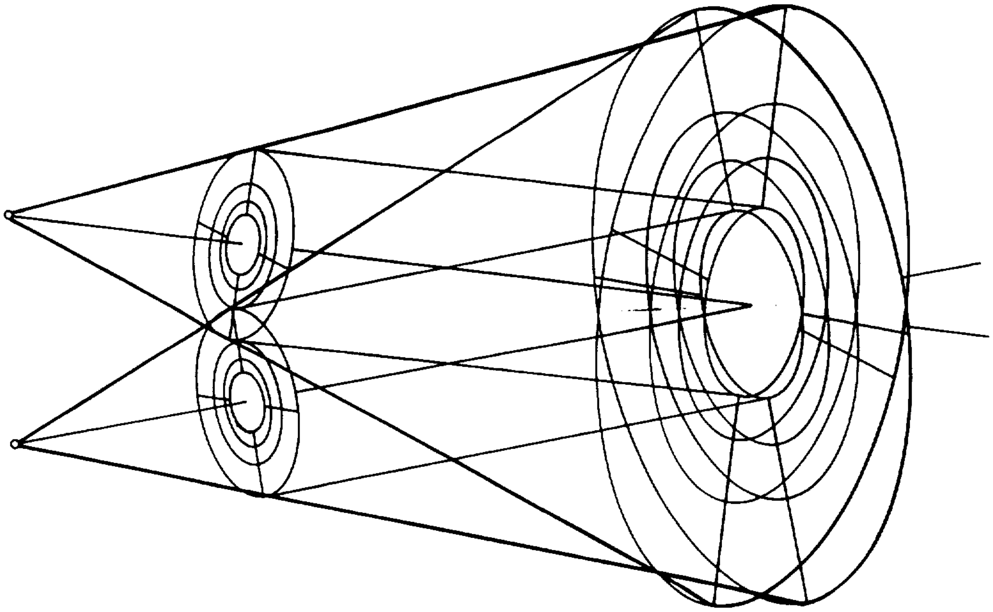


Figure 3-24. Convergent projection of log-polar retinas into visual field

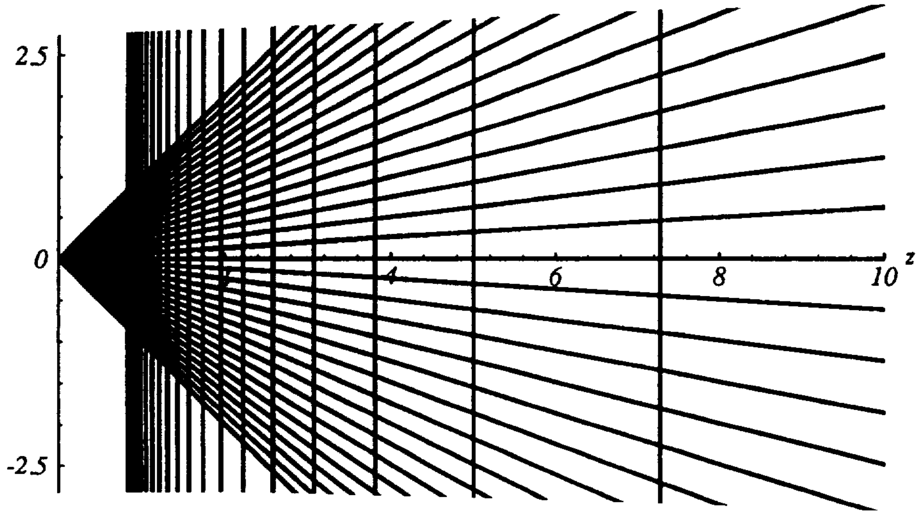


Figure 3-25. Saggital plane section for convergent Cartesian cameras

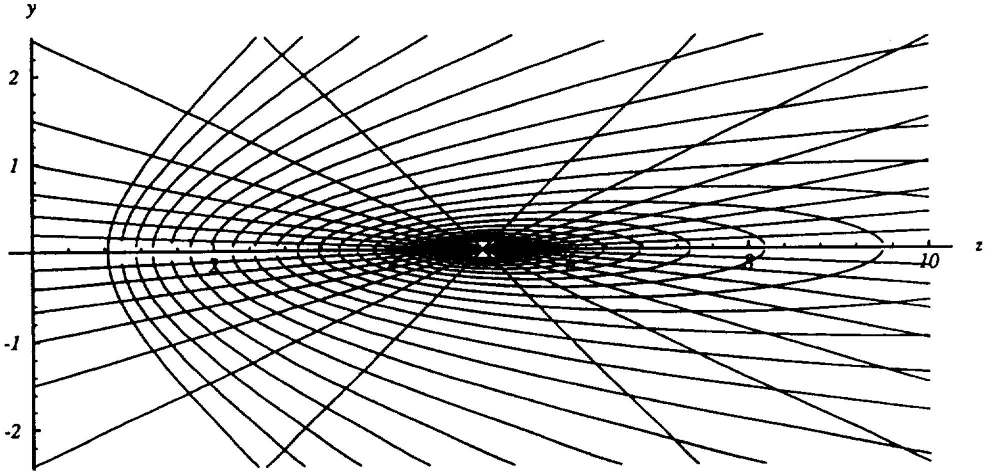


Figure 3-26. Saggital plane section for convergent log-polar cameras

3.4 Binocular Range Resolution at the Fovea of Log-Polar Retinas

We now quantify the advantage of the high-resolution log-polar fovea over the fixed uniform-resolution Cartesian retina for measuring range, having observed qualitative illustrations in the cross-sections and moiré patterns of the preceding paragraphs. The following discussion assumes that two cameras are symmetrically converged on a target, a reflex state of our active camera control system to be described in sections 5 and 6. Figure 3-27 illustrates a top view of two corresponding pixels at the center of the field of view of each camera. The shaded quadrilateral in the center is a voxel expressing the position uncertainty of a 3-D point which is visible by both pixels simultaneously. We approximate this quadrilateral by a *parallelogram* to simplify the geometric analysis which follows, justifying the approximation by noting that for small angles, i.e. pixel subtense of a few milliradians, the error incurred is less than one percent; that is, pixel subtense is a differential quantity. The key observation is that the aspect ratio of such a voxel is the ratio of range Z to half-baseline, namely,

$$a = \frac{2Z}{B} . \tag{3-16}$$

as seen by similar triangles in figure 3-27. The length of the voxel is the magnitude of the range error for binocular stereo. Since this quantity is proportional to pixel subtense, we can quantify the log-polar range resolution advantage directly in terms of pixel size as follows.

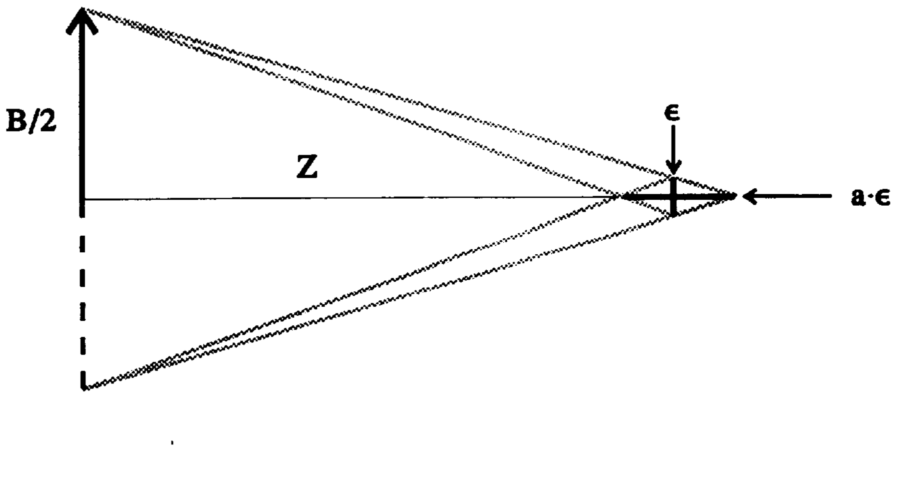
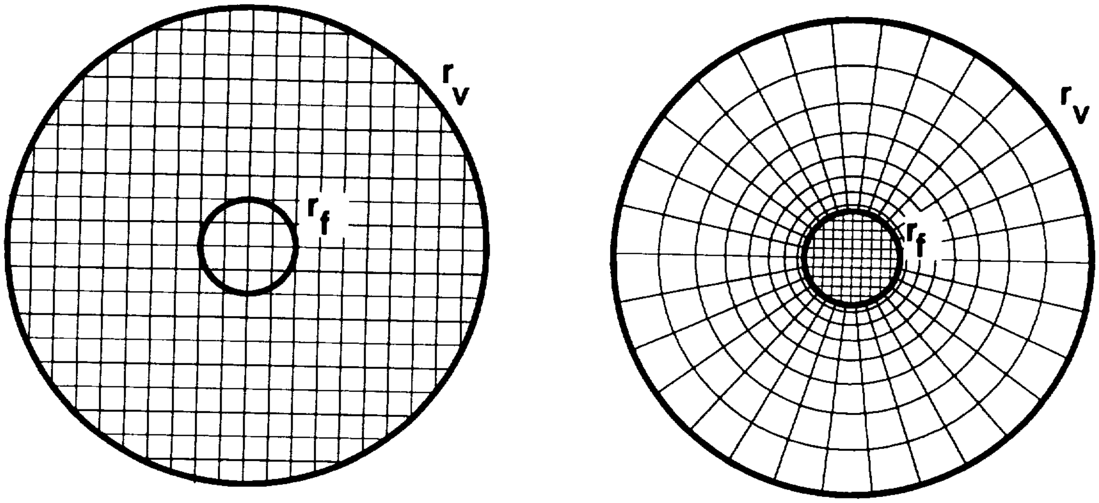


Figure 3-27. Voxel geometry at the binocular center

Figure 3-28a illustrates a uniformly tessellated (Cartesian) disk of radius r_v in pixel units, and figure 3-28b a log-polar disk of the same radius and same *total* pixel count. That is, for equitable comparison, we constrain the number of pixels to be equal in both "retinas".

Figure 3-28b depicts the fovea in Cartesian tessellation in order to allocate pixel count to disk area. Of course, any of the "designer" foveas of section 2 will have approximately the same pixel count, with more harmonious geometry.



a) uniform tessellation

b) log-polar tessellation

Figure 3-28. Cartesian vs. log-polar pixel tessellation

We now parametrize the concentration C of foveal resolution as the ratio of field-of-view radius r_v to fovea radius r_f in figure 3-21b,

$$C = \frac{r_v}{r_f} . \qquad (3\text{-}17)$$

The number of pixels in the Cartesian retina is then the area of the disk,

$$npix = \pi \; r_v^2 \quad . \qquad (3\text{-}18)$$

Now, the log-polar periphery of figure 3-21b is divided into n rays and q rings where

$$q = \frac{n}{2\pi} \log_e \left(\frac{r_v}{r_f} \right) = \frac{n}{2\pi} \log_e C \qquad (3\text{-}19)$$

by virtue of conformality and unit aspect ratio pixels [Weiman, 1988a].

Pixel count in the fovea is

$$\frac{n^2}{4\pi} \quad . \tag{3-20}$$

Thus, equating pixel count in Cartesian and log-polar retinas yields

$$npix = \pi\, r_v^2 \;=\; q \cdot n + \frac{n^2}{4\pi} \;=\; \frac{n^2 \log_e C}{2\pi} + \frac{n^2}{4\pi} \tag{3-21}$$

whence

$$n = \frac{2\pi\, r_v}{\sqrt{1 + 2\,\log_e C}} \tag{3-22}$$

Now, the foveal radius in pixel count for figure 3-28b is

$$\frac{n}{2\pi} = \frac{r_v}{\sqrt{1 + 2\,\log_e C}} \tag{3-23}$$

and the radius in pixel count of the same region in the Cartesian counterpart is simply its radius,

$$r_f = \frac{r_v}{C} \tag{3-24}$$

by virtue of unit pixels in that domain. Dividing these two pixel counts gives the resolution ratio

$$R_f = \frac{C}{\sqrt{1 + 2\,\log_e C}} \quad . \tag{3-25}$$

That is, ***equation 3-25 expresses the ratio of improvement in foveal resolution, and hence improvement in range resolution, for log-polar foveas over Cartesian foveas, for any given pixel count per field of view.*** It is interesting to note that this quantity is *independent of absolute resolution* (total pixel count), depending only on the ratio of field of view to fovea. Figure 3-29, which plots the value of this function, shows that despite the transcendental appearance of the equation, the relation is nearly linear with respect to C because the denominator changes so slowly. This relation is approximated by a gain in range resolution roughly proportional to the ratio of field of view to fovea, with a constant of proportionality of about 1/3. Thus, foveas 1% the diameter of a log-polar retina (i.e. C = 100) yield a 30-to-1 gain in stereo resolution over Cartesian retinas with the same number of pixels.

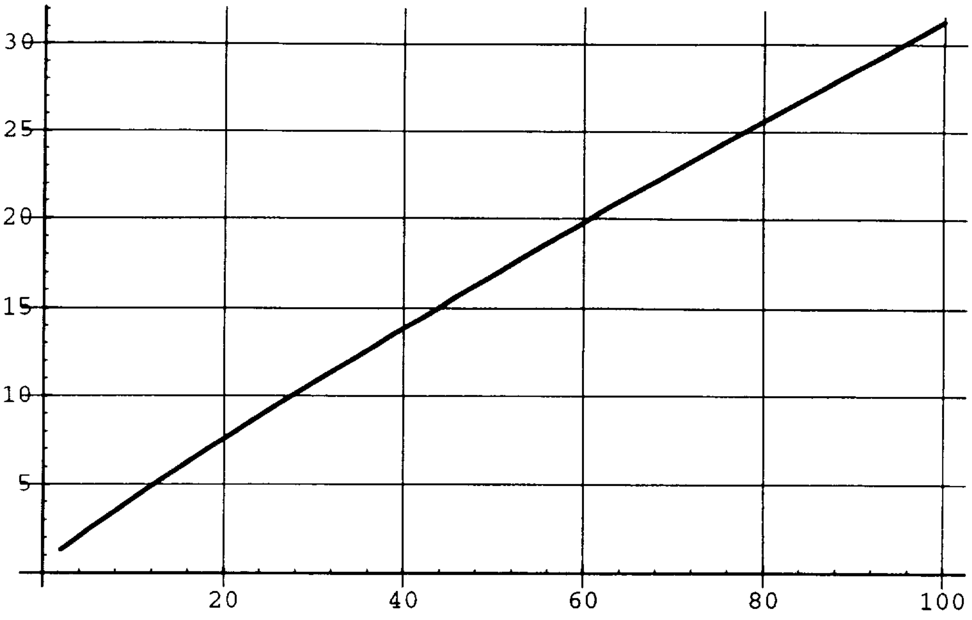


Figure 3-29. Range resolution ratio per fovea size relative to field of view

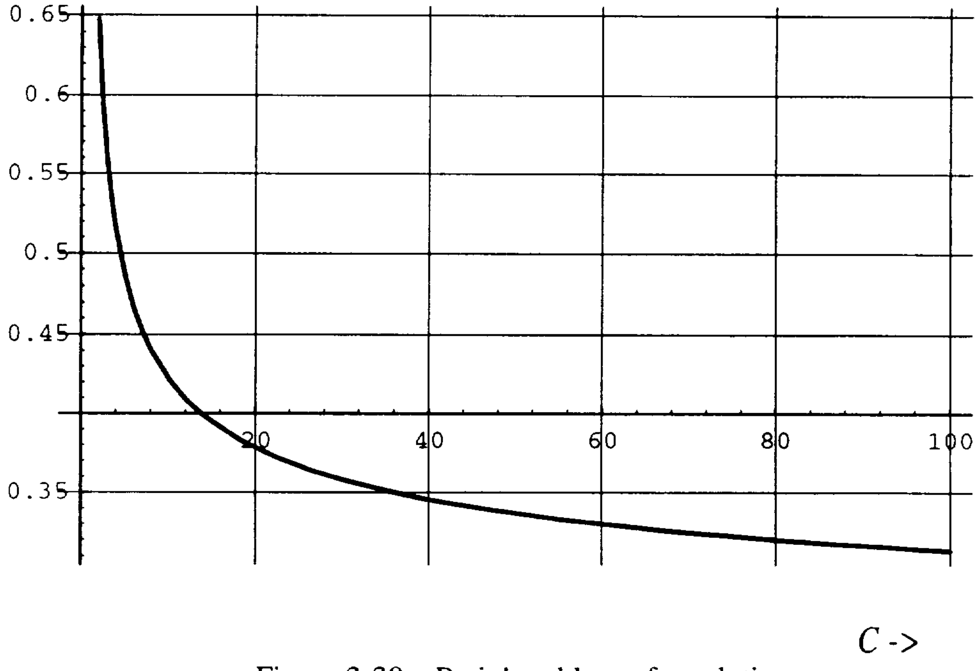


Figure 3-30. Peripheral loss of resolution

Surprisingly, the penalty paid in decreased peripheral resolution is proportionally far less than the gain in foveal resolution. Equation 3-26 gives the ratio of log-polar to Cartesian pixel diameter at the extreme periphery,

$$R_v = \frac{R_f}{C} = \frac{1}{\sqrt{1+2\log_e C}} \quad . \tag{3-26}$$

This function is plotted in figure 3-30. Note that for comparable values of C figure 3-29 exhibits more extreme behavior. Another way to characterize this phenomenon is to express the position of the "crossover" radius r_x, where Cartesian and log-polar pixels are of equal size in both retinas.

The equation for this radius is

$$r_x = \frac{r_v}{\sqrt{1+2\log_e C}} \tag{3-27}$$

Figure 3-31 plots r_x/r_v . Note the gradual decrease in r_x which ranges from 40% of r_v at C = 10 to about 30% of r_v at C = 100.

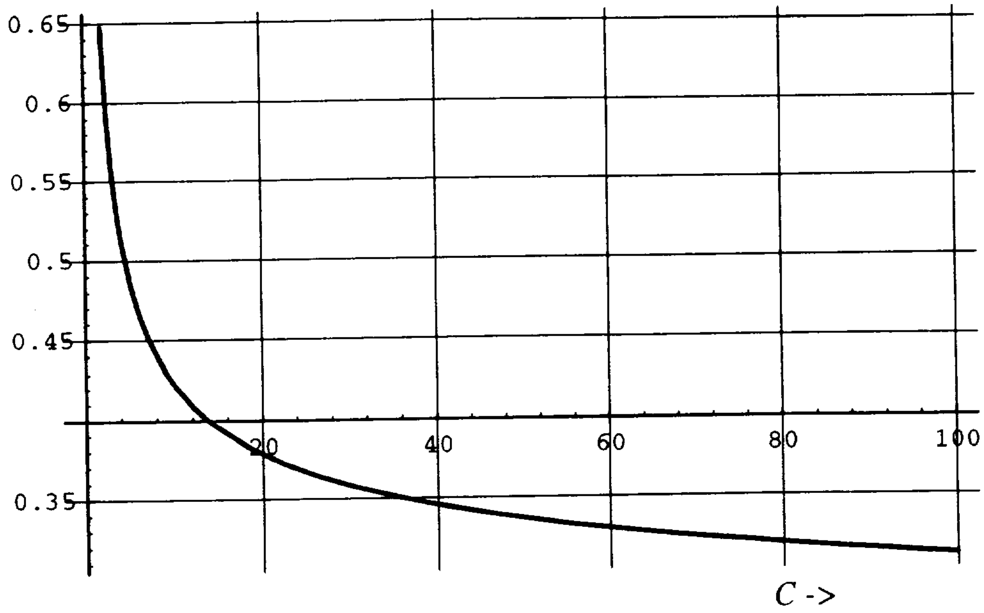


Figure 3-31. Crossover radius

Let us now examine some practical examples using parameters in the range of typical applications. For example, consider a binocular camera system based on 512 pixel diameter (r_v = 256) image planes with 90° fields of view. For a baseline B of 25 centimeters, and a vergence distance of 2 meters, equation 3-16 tells us that voxel aspect ratio is 16-to-1.

Pixel cross section is simply range times angular subtense in radians,

$$\frac{\pi}{2\cdot 512} \times 2\ meters = 6\,mm \tag{3-28}$$

whence range uncertainty is

$$16 \times 6\,mm = 9.6\,cm \tag{3-29}$$

Now consider a log-polar retina with the same total pixel count in the disk of radius $r_v = 256$ and a fovea whose diameter is 1% of r_v, i.e., $r_f = 26$, or $C = 100$. From equation 3-25, the density ratio is 31.3-to-1, i. e., better than 30-to-1 improvement in range resolution to 3.26 mm. From equation 3-22 log-polar ray number n is thus 503. The crossover radius is $r_x = 80$ from equation 3-27.

Another interesting example is the human eye whose cone (color sensing pixels) distribution can be closely modeled as log-polar, with a value of C close to 100 (fovea diameter is about 1% of retinal diameter) which yields a 31-to-1 improvement in stereo resolution over a uniformly tessellated retina with the same number of photoreceptors. With cone spacing of approximately 30 arc seconds and binocular baseline of 63 millimeters, the geometric length of a voxel is .4 millimeters (400 microns) for close eye-hand coordination (threading a needle, or delicate assembly at a range of 30 centimeters, elbows bent). Voxel width is approximately 42 microns. By way of example, fine thread is about 100 microns in diameter and human hair is in the small multiples of tens of microns.

Now, having shown that concentrating pixels in the fovea gives nearly two orders of magnitude improvement in range resolution for binocular convergent cameras, in the next section we show how another order of magnitude can be gained by using a 2-D signal processing filter capable of resolving phase to sub-pixel precision for binocular disparity.

4.0 GABOR FILTERS FOR STEREO DISPARITY MEASUREMENT

The preceding section discussed the geometry of binocular stereo, and the significant improvement in stereo resolution produced by log-polar pixel tessellations which concentrate high resolution in the center of the field of view. In this section we describe an efficient computational structure for performing disparity measurements to subpixel accuracy. We define a new discrete version of the Gabor filter which can be iterated over the image plane. This architecture could replace traditional number crunching vision preprocessors with an array of 2-D signal processors operating in parallel at very high speeds. In fact, the filter and architecture are inspired by the structure of the primate visual cortex and its neuronal components, the simple and complex cells discovered by Nobel Prize winners [Huebel and Wiesel, 1962]. Numerous models of this structure have been described and studied in the literature. Here we offer a minimalist version which shows promise as a practical processor in a "silicon cortex" for robotic vision.

4.1 Background: The Correspondence Problem and Active Vision

The *correspondence problem* may be expressed as the question:

> "Given the image-plane location of a visual component as seen from one viewpoint, what is its image-plane location as seen from another viewpoint?"

Once the question is answered, triangulation via camera-model equations finds the 3-D locus in the environment which generated both images. Both identification and localization are involved in the correspondence problem. Identification characterizes the two components in the separate images as having been generated by the same environmental object. Localization measures the difference in position (disparity) of the component in the two images once it has been identified.

Approaches to the correspondence problem range from high level object recognition to low level matching mechanisms. The high level approaches involve heuristics for global identification, but the low level methods are faster, more robust, and computationally simpler in parallel computation architectures. Low level methods are also used in nature [Julesz, 1971] and are at the heart of the Gabor filter approach discussed below. Following Julesz, Marr's work [Marr and Poggio, 1979] demonstrated local computational mechanisms for plausible assignment of correspondence, given configurations with multiple possible interpretations.

Successful low level methods typically sidestep the identification problem by applying local matching techniques. That is, without explicitly characterizing features, a measure is made of how well one image locus matches another. Numerical measures such as cross-correlation peaks are maximized where the match is best, and the position of the peak gives an accurate measure of disparity. But cross-correlation is computationally expensive. This problem has been surmounted by using Fourier transforms of both images, whose product is the Fourier transform of the cross-correlation. The computational expense of taking a Fourier transform can be avoided by optical holography.

The major drawback to optical correlation is that that the operation is not localized. Marr's approach addressed this problem by applying the zero-crossings of LaPlacians as *local* feature identifiers. A shortcoming of this technique is that slight deformations in image brightness gradients move the zero-crossings substantially, introducing noise into the measurement. Robustness was improved by [Nishihara, 1984] who built real time hardware for correlating signs of LaPlacians, reducing zero-crossing noise by effectively integrating over extended areas where the sign is constant. An ingenious method related to cross-correlation was devised by [Yeshuran and Schwartz 1989a] using the logarithm of the power spectrum (so-called cepstrum). In his dissertation [Coombs 1992] analyzes the algorithm as an approximation of broad band filtering followed by correlation.

Local methods suffer the dilemma of scale choice for both the reach and size of the window. Larger windows accommodate larger disparities and improve the correlation signal-to-noise ratio, provided the viewed object is characterized by constant disparity. As this condition is less likely the larger the window, smaller windows more faithfully depict disparity, but with more noise. Worse yet, if the size of the window is less than the magnitude of the disparity, correspondence will not be achieved. An adaptive method by [Kanade et al, 1994] controls window size via feedback from measures of local disparity coherence. This requires computational mechanisms above the simple local neighborhood operators, a complexity which we wish to avoid.

David Coomb's landmark dissertation [Coombs, 1992] expressed a principle for overcoming the preceding problems via active vision, which we take the liberty of paraphrasing below:

Active vision simplifies the correspondence problem by reducing disparity to zero via tracking, eliminating the identification component of the problem so long as tracking is successful, and reducing the disparity measurement problem to small deviations from zero.

Our objective in designing a disparity measuring operator was to minimize computation and maximize performance based on the active vision premise cited above, namely, restricted to small deviations from zero disparity. By "operator" we mean a local filter applied to windows of pixels, whose output is a number signifying disparity. The Gabor filter has recently been demonstrated to be a superb operator for disparity measurement, [Fleet, 1992], [Fleet and Jepson, 1993], [Braithwaite and Beddoes 1994], [Tölg, 1992]. Specifically, [Theimer and Tölg 1992] have used Gabor phase for active vision vergence servoing.

A serious disadvantage to Gabor filters is computational cost. The filter window is generally much wider, e.g. 32 pixels rather than the traditional 3x3 neighborhoods for LaPlacians and Sobel operators, and aliasing may introduce erroneous measurements. We designed a "minimalist" discrete version of the Gabor filter reconciled with the finite resolution limits of the image plane pixel array. Our criteria were to minimize neighborhood size, avoid aliasing, maximize resolution, and completely cover the spatial and frequency domains. The result was an 8x8 filter capable of resolving subpixel disparity. Details of the derivation are given in [Weiman 1994], excerpted below.

4.2 The Continuous Gabor Filter and Phase

The Gabor filter was originally devised as a quadrature mirror filter which minimizes joint spatial/frequency domain uncertainty [Gabor 1946]. Since we know that spatial uncertainty resolution is limited by the separation distance of adjacent pixels, we design to push the frequency uncertainty limit as far as possible subject to the joint uncertainty constraint.

Figure 4-1 illustrates the real and imaginary components of the continuous Gabor filter, a sine-cosine pair damped by a Gaussian,

$$g(x;\mu,\sigma,\omega) = Gauss(x;\mu,\sigma)e^{i\omega\ x} \qquad (4\text{-}1)$$

where

$$Gauss(x;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\exp(\frac{(x-\mu)^2}{2\sigma^2}) \quad . \qquad (4\text{-}2)$$

Expanding the second term of equation 4-1 into its real and imaginary parts yields the familiar Gabor quadrature pair, illustrated in figures 4-1a and 4-1b, consisting of cosine and sine wave modulation under a Gaussian envelope,

$$g = g_c + i\,g_s \qquad (4\text{-}3)$$

where

$$g_c(x;\mu,\sigma,\omega) = Gauss(x;\mu,\sigma)\cos(\omega\ x) \qquad (4\text{-}4)$$

and

$$g_s(x;\mu,\sigma,\omega) = Gauss(x;\mu,\sigma)\sin(\omega\ x) \qquad (4\text{-}5)$$

The modulation ω is limited by the Nyquist sampling rate of pixel tessellation and σ determines bandpass. These parameters determine the design of the discrete filter shown in figure 4-2. Before leaving the continuous Gabor filter to derive its discrete counterpart, we describe the concept of phase because of its importance to hyperacuity, i.e. sub-pixel resolution of position.

A primary application of Gabor filters in artificial vision is the measurement of phase to subpixel precision (hyperacuity). A major objective of our discrete filter design is to measure phase accurately and efficiently. Phase correlates with image position, which must be precisely measured for binocular disparity and optic flow [Fleet, 1992], [Fleet and Jepson, 1993], [Sanger, 1988], [Tölg, 1992]. Fleet and Jepson have shown that phase is robust against noise, illumination variations, and small geometric distortions arising from different perspective viewpoints. We illustrate Gabor phase with the following simple example.

Consider the image of a contrast edge which is bright (intensity = 1) to the left and dark (intensity = 0) to the right of some position x_h along the horizontal axis, as illustrated in figure 4-1c, where the shaded area corresponds to the bright side. That is,

$$f(x;x_h)=1 \quad for \quad x<x_h \quad ; \quad f(x;x_h)=0 \quad for \quad x \geq x_h \quad . \tag{4-6}$$

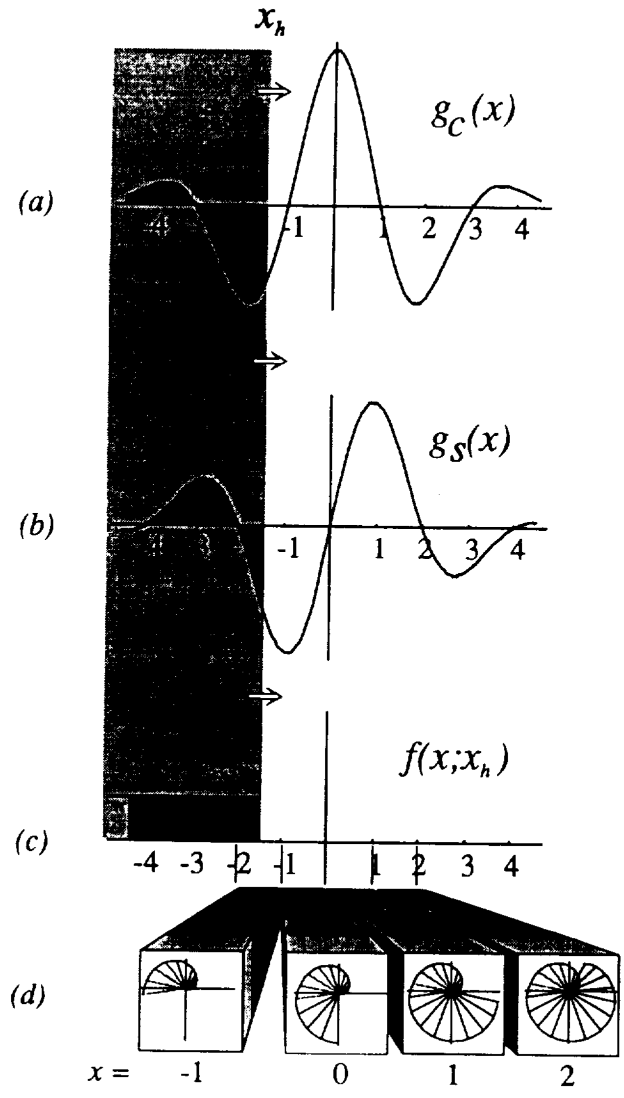


Figure 4-1. Continuous Gabor filter

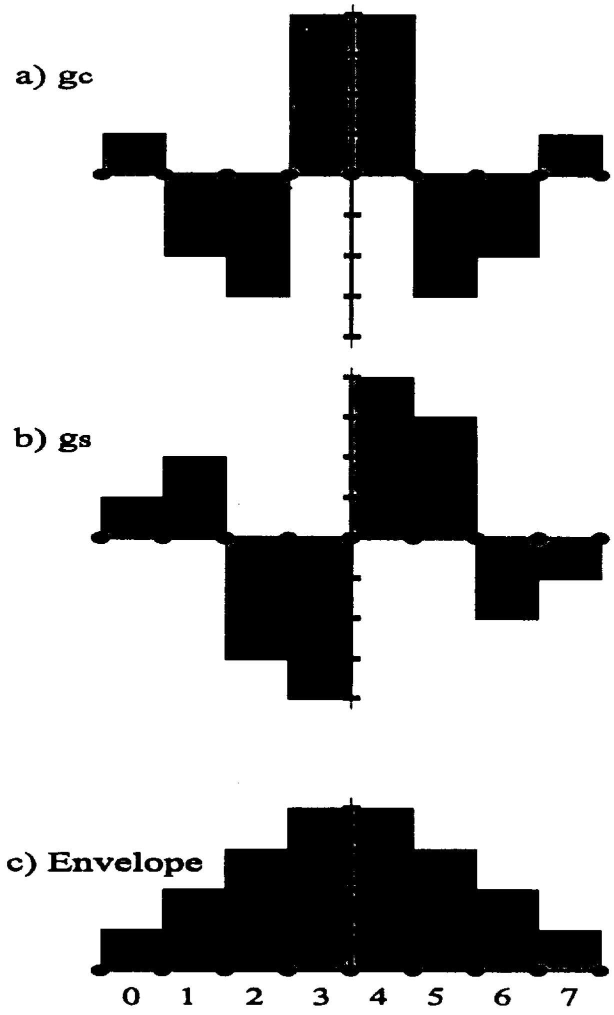


Figure 4-2. Discrete Gabor filter

Now, applying the cosine and sine components of the Gabor filter (g_c and g_s in equations 4-4 and 4-5) to the edge step function of equation 4-6 corresponds to the functionals

$$G_c(x_h) = \int_{-\infty}^{\infty} f(x;x_h) g_c(x)dx = \int_{-\infty}^{x_h} g_c(x)dx \qquad (4\text{-}7)$$

and

$$G_s(x_h) = \int_{-\infty}^{\infty} f(x;x_h) g_s(x)dx = \int_{-\infty}^{x_h} g_s(x)dx \qquad (4\text{-}8)$$

illustrated by the shading in figures 4-1a. That is, equations 4-7 and 4-8 express the outputs of filters g_c and g_s applied to edge $f(x; x_h)$. The curved outline of figure 4-3 plots the joint trajectory of G_c and G_s as x_h , the visual edge, moves across the filter window. The circular path reflects the underlying sine and cosine modulation while the spiraling to center is a consequence of the Gaussian envelope. Any given point on this trajectory corresponds to some position x_h of the edge. The spokes mark quarter-unit positions of x_n. Every fourth spoke thus represents one complete unit of position change on the x-axis, labeled in correspondence with figure 4-1. Figure 4-1d expands the view of the x-axis above; icons are miniatures of the phase trajectory of figure 4-3. The progression of icons corresponds to the phase associated with the segment in which each icon resides.

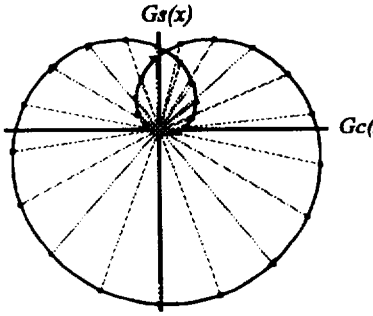


Figure 4-3 Gabor phase trajectory

Regarding the G_c and G_s axes in figure 4-3 as components of a complex number, in keeping with equation 4-3, the argument (phase angle) corresponds to the phase of the output of the quadrature filter pair.

$$\phi_h = arg(G_c(x_h) + iG_s(x_h)) = \arctan(G_s(x_h) / G_c(x_h)) \qquad (4\text{-}9)$$

Note that phase is independent of both amplitude and offset of the image step function. That is, for a stronger edge,

$$k f(x;x_h) \ ; \ k > 1 \qquad (4\text{-}10)$$

equations 4-7 and 4-8 are scaled proportionally by k, and their quotient in 4-9 cancels k. DC offsets integrate to zero, because the Gabor filter integrates to zero, so nothing is contributed by G_c or G_s.

For Fourier transforms, phase is linear with position shift Δx, and sensitivity is proportional to spatial frequency ω,

$$f(x-\Delta x) \Leftrightarrow e^{-i\omega\,\Delta x}\,F(\omega) \qquad (4\text{-}11)$$

courtesy of the Fourier shift theorem. The circular portion of the trajectory in figure 4-3 shows that this relationship holds fairly well at the center of the Gabor filter, but not at the ends. The Gabor filter has been characterized as a localized Fourier transform. The distribution of spokes in the lower half plane indicates that transit of one half wavelength in the spatial domain corresponds to approximately 180° of phase transition, and the distribution is rather uniform. For a wavelength of four units, this represents 90° phase transit per unit. Thus, even modest phase discrimination techniques yield hyperacuity measurements for binocular disparity and optic flow. We shall revisit this crucial property of phase in terms of discrete filters later, where units correspond to pixels.

4.3 The Discrete Gabor Filter and its Phase

The ultimate limit to resolution is the discrete tessellation of pixels in the image plane. Their separation dictates the highest modulation frequency possible under the Gaussian envelope of the Gabor filter; courtesy of Nyquist this frequency ω and the corresponding wavelength L are

$$\omega = \pi(perpixel) \;;\; L = 2\;(pixels) \qquad (4\text{-}12)$$

Figure 4-4 illustrates the quantized sine wave whose frequency is given in the preceding equation, normalized to unit amplitude. But, the Gabor quadrature pair requires a cosine wave as well, i.e., a quarter wave shift of the sine. For $L = 2$, this is unrealizable since the shift represents a fraction (1/2) of a pixel; subpixel weighting for such a shift would yield zero value throughout as is obvious from figure 4-4. Thus, quadrature dictates that integer valued wavelengths must be multiples of four pixels. The highest possible frequency (shortest possible wavelength) satisfying these criteria are

$$\omega = \pi/2 \;;\; L=4 \qquad (4\text{-}13)$$

as illustrated in figure 4-5. Note that quantization and symmetry reduce the sine and cosine to square waves, which simplifies computation. Dots on the x axis in figures 4-5 and 4-6 correspond to pixel boundaries.

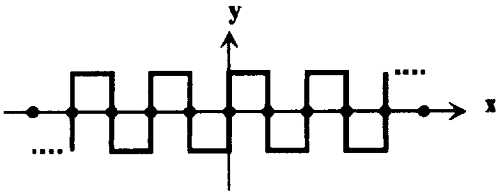
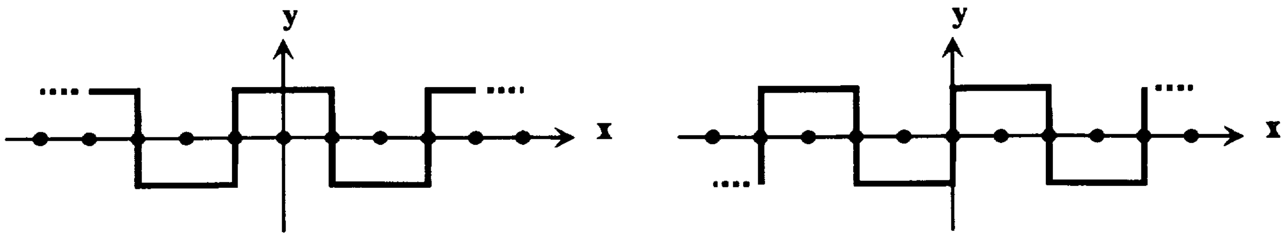


Figure 4-4 Highest possible frequency quantized sine wave



a) Cosine component

b) Sine component

Figure 4-5. Highest possible frequency quantized quadrature pair

Having derived the frequency modulation for the discrete filter, we now derive the Gaussian envelope based on bandpass requirements.

For the Gabor filter, reciprocal uncertainty constraints between image plane position and frequency plane bandpass are in exact analogy with the Heisenberg Uncertainty Principle, as expressly designed [Gabor, 1946]. Small values of sigma for the Gaussian envelope yield large bandpass which leads to undesirable aliasing; [Fleet and Jepson, 1993] cite scale space phase detection singularities arising from such aliasing. Bandpass must however be large enough to bridge the gap between levels of an image pyramid, lest the system be blind to intermediate spatial frequencies. For dyadic up- and down-sampling (doubling/halving in each dimension), the custom for image pyramids, each level of the pyramid corresponds to a jump of one octave. Thus one octave bandpass is necessary to span levels. Translating Sanger's equation 4-5 [Sanger, 1988] from frequency to spatial domains, the relation between bandpass and σ (of equation 4-1) is

$$B = \log_2\left(\frac{2\pi\sigma + L}{2\pi\sigma - L}\right) \qquad (4\text{-}14)$$

where B is bandpass in octaves and L is wavelength of the underlying Gabor sine and cosine filters. Thus, bandpass of one octave implies that the quantity within parentheses equals 2, whence

$$\sigma = .48L \qquad (4\text{-}15)$$

The filters illustrated in figure 4-1 exhibit precisely this relation between wavelength L and Gaussian spread σ. Note that the modulated cosine 1a) resembles a cross section of the widely used "Mexican hat function" defined as the difference of Gaussians or the 2nd derivative of a Gaussian (LaPlacian of Gaussian). The companion b) closely resembles the 1st derivative of a), in keeping with Koenderink's jet bundle model [Koenderink et al, 1987].

Now, using the quadrature pair of figure 4-5 for constructing the discrete filter, wavelength L is 4 pixels. Substituting this value into equation 4-15 yields

$$\sigma = .48 x 4 = 1.92 (pixels) \tag{4-16}$$

for the Gaussian modulation. Figure 4-6 illustrates the Gaussian for this value of σ quantized per pixel by integrating the continuous curve over unit (single pixel) intervals.

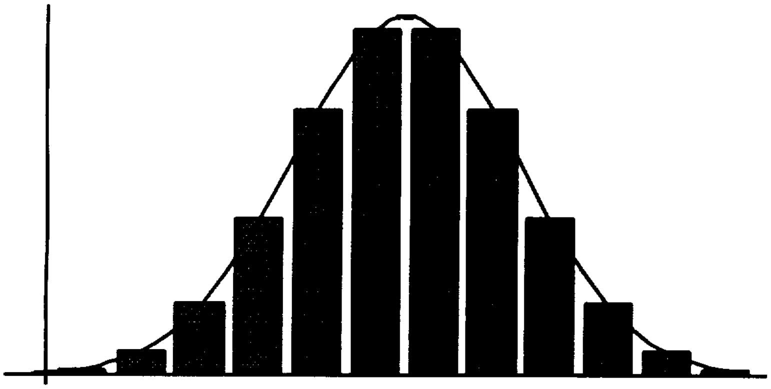


Figure 4-6. Quantized Gaussian for σ = 1.92 pixels

Note that contributions beyond four pixels from the center are very small. Thus an 8 pixel window is both necessary and sufficient to accommodate the derived values of L and σ (equation 4-16). Figure 4-2 illustrates the quantized values of the quadrature pair of this discrete Gabor filter generated by applying the envelope of figure 4-6 to the (square) sine and cosine waves of figure 4-5. Constructively, this operation can be regarded as flipping the signs of alternating pairs of coefficients of the quantized Gaussian of figure 4-6.

The discrete filter offers high precision positional measurement via phase just as the continuous Gabor filter. Figure 4-7 illustrates the phase trajectory of the discrete Gabor filter of figure 4-2 as a vertical edge is moved through the window, analogous to figure 4-3. A striking difference between the continuous and discrete cases is that the latter displays straight line segments rather than a circular trajectory. Examination of figure 4-2 reveals why. As an edge moves across any single pixel, the contributing area under the filter grows linearly for both the sine and cosine component. Every pixel boundary is accompanied by a sign transition in either the sine or cosine filter coefficient. Thus, each linear segment in the phase trajectory corresponds to the transit of an edge across one pixel. It is evident that subpixel motions can easily be measured with high precision with such a mechanism (hyperacuity).

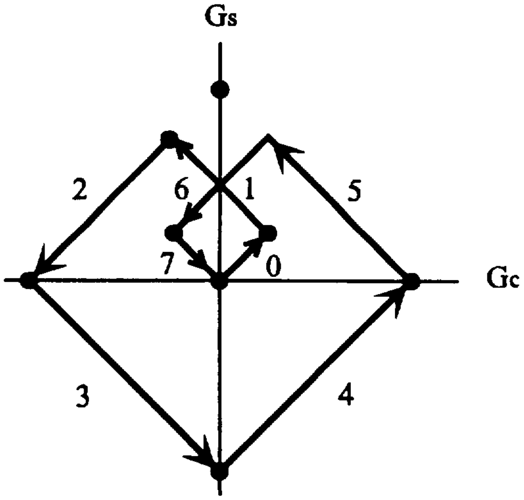


Figure 4-7 Phase trajectory of discrete Gabor filter

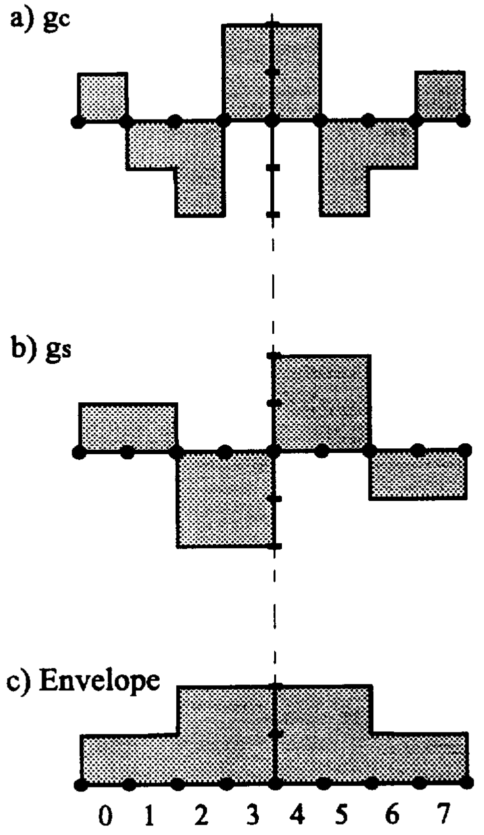


Figure 4-8. Ideal discrete phase filter

Two minor imperfections are manifested in figure 4-7 which prevent this filter from being an ideal phase detector. One is the small inscribed diamond shaped trajectory near the origin, which represents the "ramping up" of filter energy in the outer pair of entry pixels (numbered 0 and 1 in figure 4-2) at the beginning of the trajectory and ramping down at the pair of exit pixels (labeled 6 and 7 in figure 4-2) at the end of the trajectory. Aside from these four pixels, linear segments for the central four pixels are nearly segregated by quadrant. The fact that they are only *nearly* so comprises the second imperfection. Segments labeled 2 and 5 (corresponding to so-numbered pixels in figure 4-2) are 25% shy of filling the quadrant, unlike segments 3 and 4 which fill their respective quadrants. If pixel transits were exactly segregated by quadrant, phase computation would be extremely simple. Filter signs would point to pixel, and phase would be linearly related to subpixel position.

We now "reverse engineer" the discrete Gabor filter of figure 4-2 by adjusting its coefficients in order to establish a linear relationship between phase and edge position and segregate pixel transits by phase quadrant. In so doing, we wish to preserve as much as possible the frequency and bandpass of the Gabor filter. Thus, the underlying sine and cosine components and Gaussian envelope should be disturbed as little as possible.

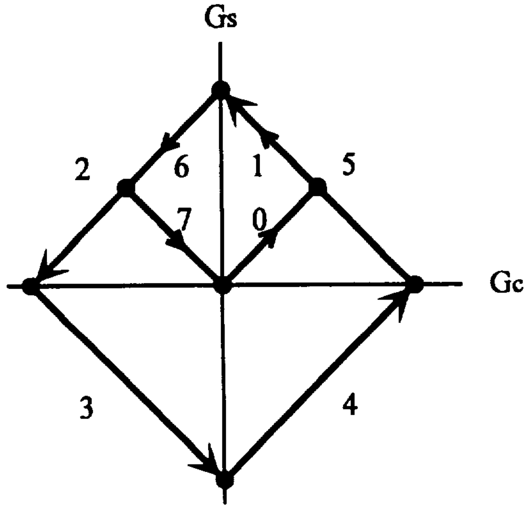


Figure 4-9. Phase trajectory of ideal phase filter

Figure 4-8 illustrates an ideal phase filter which fulfills the above requirements. Figure 4-8c is the discrete version of its Gaussian envelope, modified so that the outer shoulders (pixels labeled 0, 1, 6, 7) are constant at half the height of the central plateau (pixels 2, 3, 4, 5). Figures 8a and 8b respectively give g_c and g_s, applying the modulation of figure 4-5. By weighting the central four pixels equally, the phase trajectory for the four middle pixels is a perfect square centered at the origin with vertices on the principal coordinate axes, as illustrated in figure 4-9. The half-weights of the outer shoulders ramp up filter energy to exactly reach the first vertex of the square. Figure 4-9 illustrates the perfect association between phase trajectory and position of x_h . Linear segment labels correspond to pixel numbers, and progression along a segment is linear with x_h . Arrows correspond to left-to-right motion of the edge x_h.

Since the signs of the two filter components alternate on pixel transitions, each phase quadrant corresponds to a single pixel. Thus the pixel index of an edge can be derived from the combinations of the signs of the outputs of the two filters as follows.

Labeling the pixels 0 through 7 from left to right (see figure 4-8) and quadrants 0 through 3 clockwise, the sign transition of quadrants is :

pixel #	0	1	2	3	4	5	6	7
quadrant	0	0	**1**	**2**	**3**	**0**	1	1
sign (g_c , g_s)	(+, +)	(+, +)	(- , +)	(- , -)	(+ , -)	(+ , +)	(- , +)	(- , +)

Table 4-1. Phase quadrant relations for ideal phase filter

The symmetry of figure 4-9 shows that this filter can be used as an ideal "phase only" filter if we modify the definition of phase to correspond to linear segment traverse, namely,

$$\phi\ (x_h) = \frac{G_C - G_S}{|G_C| + |G_S|} \tag{4-17}$$

for even quadrants and

$$\phi\ (x_h) = \frac{G_C + G_S}{|G_C| + |G_S|} \tag{4-18}$$

for odd quadrants. Note that this filter is invariant under brightness shifts because each filter is immune to DC changes, and invariant under brightness scaling because scale factors cancel in numerator and denominator. The values of this phase function range exactly linearly with edge subpixel position x_h from -1 to 1 within each quadrant (pixel). The phase value 0 exactly bisects the pixel. The denominator is constant over the middle four pixels of the filter window. Thus, the complexity of subpixel phase computation is simpler than the continuous case, and exactly linear with position in the image.

Having established the advantageous properties of the discrete Gabor filter, we now extend its definition to cover the 2-D image plane.

4.4 Generalization of the Discrete Gabor Filter to 2-D

We derive the discrete 2-D Gabor filter by abbreviating Daugman's landmark paper [Daugman, 1985] which generalized continuous Gabor filters from one dimension to two for application to a model of processing in the visual cortex. Extend the Gabor filter to two dimensions by modulating the 1-D filters with a Gaussian in the orthogonal direction,

$$g_c(x,y;\mu_x,\mu_y,\sigma_x,\sigma_y,\omega_x) = g_c(x;\mu_x,\sigma_x)Gauss(y;\mu_y,\sigma_y) \qquad (4\text{-}19)$$

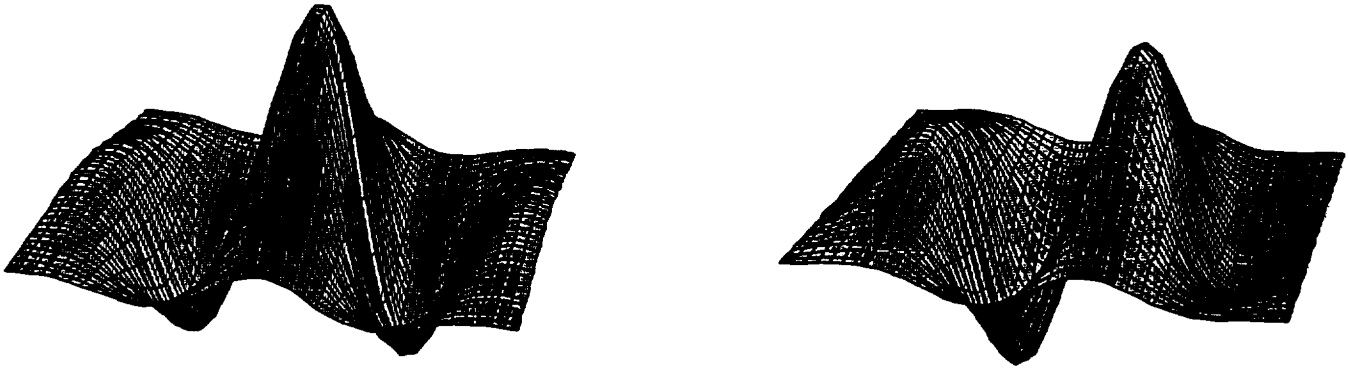
$$g_s(x,y;\mu_x,\mu_y,\sigma_x,\sigma_y,\omega_x) = g_s(x;\mu_x,\sigma_x)Gauss(y;\mu_y,\sigma_y) \qquad (4\text{-}20)$$

as illustrated in figure 4-10.

Cross sections parallel to the x-axis are scaled 1-D Gabor filters thereby extending, by linearity, the generalization to 2-D of beneficial 1-D signal processing properties regarding magnitude, energy and symmetry. Cross sections of the 2-D Gabor filter parallel to the y-axis are scaled Gaussians. The 2-D generalization has several important consequences for image analysis. One is to improve signal-to-noise ratio in the detection of oriented edges. Integration along wave crests or troughs cancels out shot noise along the edge. Noise variance is thereby reduced proportionally to cross-wave sampling volume of the filter (standard σ/n property of statistical sampling theory). Additionally, Gaussian damping across the wave eliminates sinc function ringing in that direction which would result from rectangular windowing. And finally, Daugman shows that cross-wave damping constrains the edge-orientation tuning of the filter. Such constraint is necessary if the filter is to respond appropriately to 2-D edges.

Figure 4-2 represents a cross section of a discrete filter along the direction of the wave. The 2-D extension is effected by Gaussian modulation in the orthogonal direction (equations 4-19 and 4-20). Daugman shows that orientation tuning is roughly proportional to the value of σ_y. In our experiments, σ_y values which are approximately equal to σ_x afford sufficient tuning to cover all possible orientations with a set of four filters oriented in intervals of $\pi/4$. By the same argument as surrounding figure 4-6, an 8x8 window is required. Figure 4-12 illustrates the 2-D extension of figure 4-2c, to a bivariate discrete Gaussian. Figure 4-11 applies this "ziggurat" function to the sine-cosine pair of figure 4-5 to generate the 2-D Gabor filter extended from figure 4-2.

Now consider global coverage of the image plane at all possible orientations and spatial frequencies by discrete Gabor filters. Starting with orientation, we can cover horizontal and vertical orientations by transposing indices in the discrete Gabor filters, i. e., rotating them 90°. We have found that a 45° filter is also necessary because horizontal and vertical filters do not respond well to diagonal edges due to the coarseness of quantization. However, using all four of these orientations (horizontal, vertical, right diagonal, left diagonal), edges of arbitrary orientation can be faithfully sensed as energy-weighted sums of these four filter outputs.



a) cosine component, g_c

b) sine component, g_s

Figure 4-10. Continuous 2-D Gabor filter

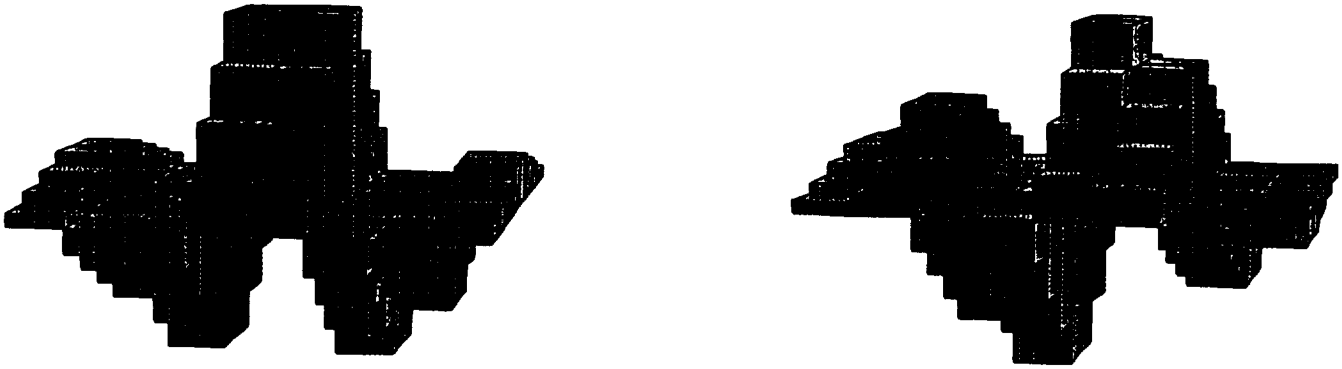


Figure 4-11. Discrete 2-D Gabor filter

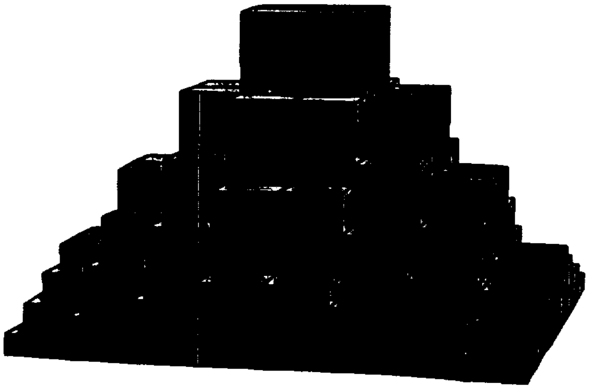
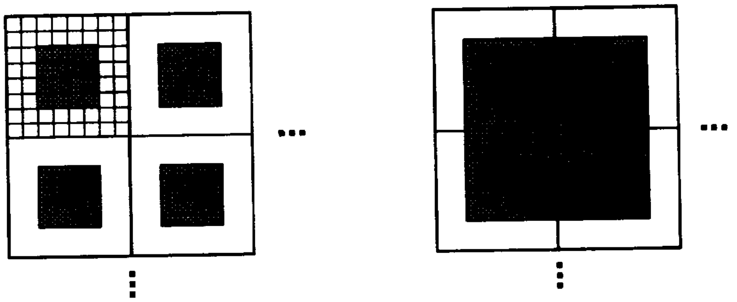


Figure 4-12. "Ziggurat" function - 2-D discrete Gaussian envelope

Image plane coverage could be achieved by square tessellation of adjacent 8x8 blocks, each covered by the four orientations of the Gabor filter pairs described above, as illustrated in figure 4-13a. However, the tapering energy profiles of the filters (segments 0, 1, 6, 7 in figures 4-7 and 4-9) would leave weak zones at the borders of such tiles (white zones in figure 13a). Hence, we overlap the blocks by 50% in both dimensions as shown in figure 4-13b which indicates neighborhood centers as diamonds. Thus the central 4x4 block gives perfect phase readings for an edge, and the entire image plane is covered by such tiles. Figure 4-14 schematically illustrates a tessellation of discrete Gabor filters covering the image plane using the four orientations (0, π/4, π/2, 3π/4) which span 360°.



a) Adjacent 8x8 blocks b) 50% Overlapping 8x8 blocks

Figure 4-13. Tessellation of 8x8 2-D Gabor filter neighborhood

A number of dyadic sampling schemes are available for pyramiding. [Burt, 1988] describes efficient hardware which can compute Gaussian up- and down-sampling over 5x5 neighborhoods at extremely high speeds. Whatever technique is used, the discrete Gabor filter can be applied as the fundamental kernel at any level of the pyramid.

Now consider frequency spectrum completeness. By virtue of the choice of σ_x = 1.92 pixels for the (nearly) Gaussian envelope, bandpass for the filter is one octave centered at 1/4 cycles per pixel. Any dyadic pyramiding process scales this band by exactly one octave, so that all frequencies are covered, at half the frequency resolution for each octave. Furthermore, superimposed frequencies are synchronized at the center of the window so that all frequency components of a step edge are identically zero-phased when the edge is centered in the window.

Total operation count per pixel at a single layer of the discrete Gabor filter pyramid can be tallied as follows. Four Gabor filter pairs span all orientations over an 8x8 pixel window, consuming 64x8 multiply-add operations per window. Doubling the geometric density of this operation in both dimensions according to figure 4-13b yields 2,048 multiply-adds per window.

Dividing by the 64 pixels in the domain yields 32 operations per pixel. For binocular stereo, double the number of image planes, which results in 64 operations per pixel. This is exactly the number of operations required for cross-correlation of two 8x8 windows. Some computational advantage for the discrete Gabor filter derives from the fact that half the multiplicands are small fixed integers rather than image data as is the case for cross-correlation. In VLSI design, more significant advantages arise from the simple sign changes and local stripe routing of data flow. Far more striking advantages derive from the comprehensive decomposition of image data into fundamental components such as spatial frequency, orientation, and phase. This "demux" operation distills the information relevant to a task, discarding the irrelevant. The effect is to considerably reduce data flow at higher processing levels. In the next subsection we give an example of Gabor phase which represents 128-to-1 compression of raw image data in the task of measuring binocular disparity for stereo vision.

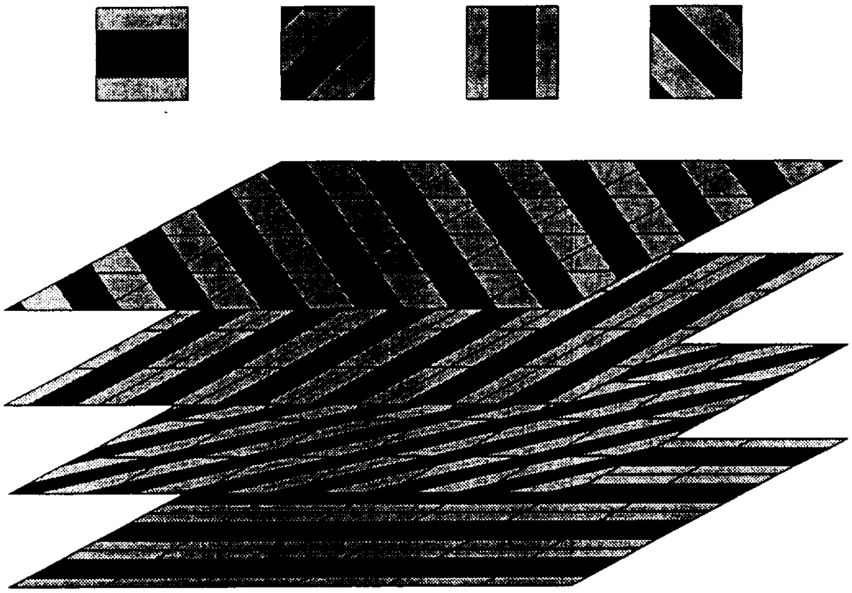
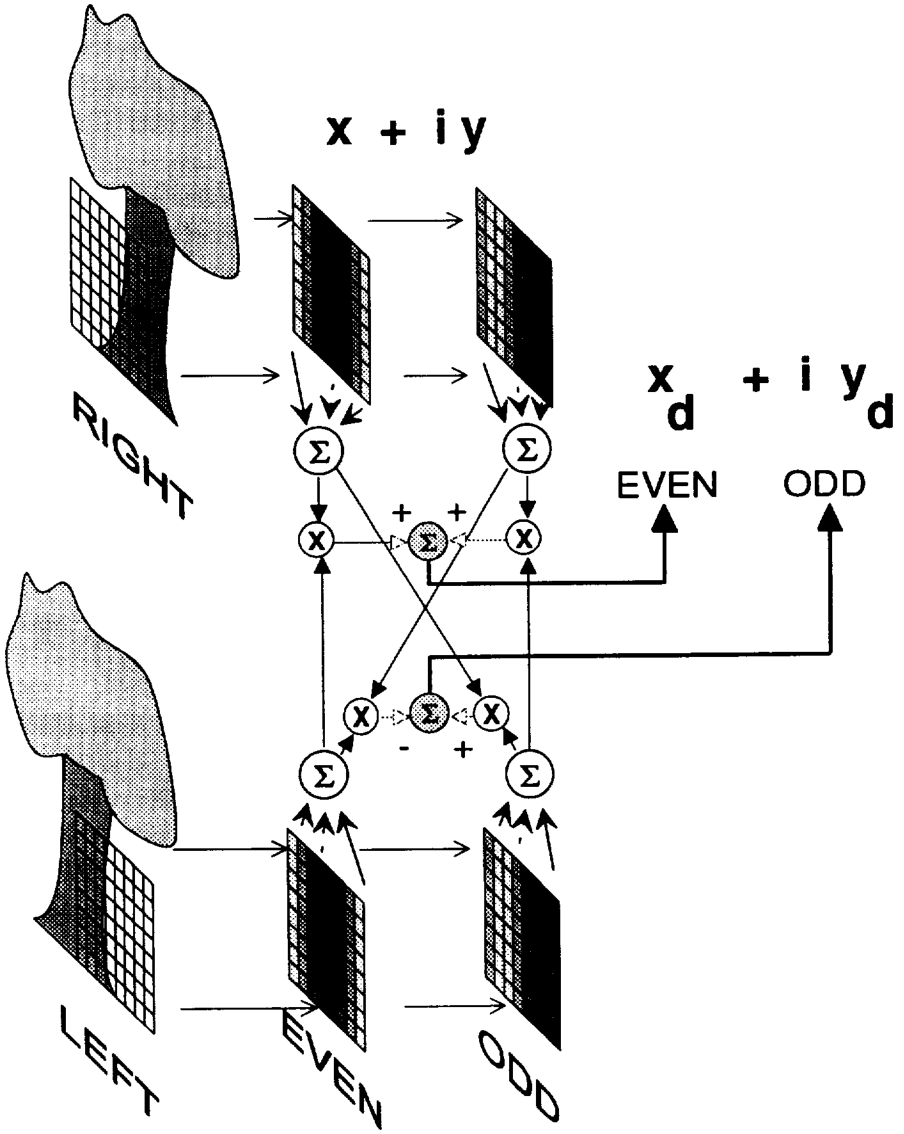


Figure 4-14. Tessellation of oriented discrete Gabor filters.

The array of discrete Gabor filters system presented here can be regarded as a model for certain structures of the visual cortex, with Gabor filters as simple cells and the left- and right-view pairing as ocular dominance column striping. This suggests searching for biological analogs of phase detection circuitry such as that shown in figure 4-15b. [Pollen, 1982] has demonstrated the existence of Gabor-like quadrature pair neurons in the visual cortex. A model [Ohzawa, 1990] of binocular disparity detection is based on the energy of such pairs. The discrete Gabor filter model presented in our report suggests a search for cells which respond to *phase* downstream from these quadrature pairs.

4.5 Gabor Filter Phase for Binocular Stereo

Figure 4-15 illustrates application of the discrete Gaussian filter to binocular disparity. In figure 4-15a a tree trunk is depicted as seen from two viewpoints through 8x8 pixel windows in each of the image planes. One image is thus shifted with respect to the other, reflecting the disparity between the two views. The pixel values of each window are input to discrete Gabor filters shown as shaded 8x8 patches at the top and bottom of figure 4-15b.



a) image disparity   b) phase disparity circuit   c) disparity signal

Figure 4-15. Phase disparity

The cosine and sine components (labeled *EVEN* and *ODD* respectively) of each filter can be treated as real and imaginary components, respectively, of a complex number (as in equation 4-3) whose argument is the phase of the filter. That is, for left- and right-view filter outputs,

$$z_L = x_L + i\,y_L \quad ; \quad z_R = x_R + i\,y_R \ . \tag{4-21}$$

the respective phases are

$$\phi_L = \arg(z_L) \ ; \ \phi_R = \arg(z_R) \tag{4-22}$$

and phase disparity is

$$\phi_d = \phi_L - \phi_R \ . \tag{4-23}$$

Recall that phase is proportional to pixel position of the edge, scaled according to the phase diagram of figure 4-9 which can be regarded as an Argand diagram of the complex filter output. The quadrant in which the filter pair output lies determines the pixel in which the edge is located, and position within the quadrant is a linear function of sub-pixel position. For two views, two Argand diagrams yield two arguments, whose difference corresponds to the disparity (equation 4-23).

Phase disparity can also be computed *directly* by exploiting the complex number model, using the property that the argument of the quotient of two complex numbers is the difference of the arguments of the components,

$$\arg(z_L \,/\, z_R) = \arg(z_L) - \arg(z_R) \ . \tag{4-24}$$

The quantity on the right is the difference in phases of left and right images from the preceding paragraph (equation 4-23). The quotient on the left is simply

$$\frac{z_L}{z_R} = \frac{x_L x_R + y_L y_R + i(x_R y_L - x_L y_R)}{x_R^2 + y_R^2} = \frac{x_d + i\,y_d}{x_R^2 + y_R^2} \tag{4-25}$$

a complex number whose numerator $z_d = x_d + iy_d$ is expressed by combining filter outputs as illustrated in figure 4-15b. Phase disparity is then

$$\phi_d = \arg(z_d) \tag{4-26}$$

by equation 4-24.

Thus by mere *arithmetic*, not programming, disparity can be generated as a hard-wired output of the circuit shown in figure 4-15b. The measure ϕ_d is recognized as the cross-correlation phase [Kuglin & Hines, 1975]. That is, consider the Gabor filter as a local Fourier transform. Cross-correlation in the image domain corresponds to quotient (product with conjugate, inversely scaled by magnitude) in the Fourier domain, so the phase of the quotient is the phase associated with the disparity shift. This measure is far more sensitive than magnitude correlation; [Oppenheim et al, 1981] notes that the autocorrelation function for phase-only signals will always be an impulse, i.e., much sharper than energy cross-correlation.

The phase-only representation of disparity compresses image data by two orders of magnitude while retaining pure stereo information, devoid of imagery. The compression path starts with 128 bytes of raw data, i.e., two (left- and right-view) 64 pixel sub-images. The horizontal disparity measure for a pair of 8x8 windows at the elevation of the horopter can be represented by a single byte, i.e. 128-to-1 compression. This 8-bit precision is based on the additive contribution of 8-bit image data pooled in 32 pixel bins of fixed sign in the operation of the discrete Gabor filter.

The span of the discrete Gabor filter (4 central pixels of the ideal phase filter of figure 4-9) determines the depth-of-field for binocular stereo measurement, which is to say, four voxels. The number of significant digits in the disparity byte determines range resolution within the depth-of-field. Refer now to the robot vision example given at the bottom of page 3-27, with 25 cm binocular baseline between cameras, converging on a target two meters away. Voxel resolution under this configuration was cited as 3.26 mm; thus four voxels yields 13 mm depth of field for the discrete Gabor filter. Eight bits of resolution for the disparity measurement over this range corresponds to a range resolution limit of 50 microns (13mm/256) at the horopter! A more conservative estimate might regard the least significant bit as noise, increasing the range resolution to 100 microns. Under this assumption, we conclude our stereo discussion with an assessment of the gain in range resolution compounded from log-polar concentration of foveal resolution and Gabor filter phase subpixelling, as follows. Recall that log-polar concentration in the example on page 3-28 decreased voxel size by a factor of 31.3-to-1 in the fovea for a choice of n = 503 rays. Gabor phase disparity resolution of 7 bits divides the resulting voxels into 32 (128/4) subpixels over the four voxel range of the filter (Aside: this value of hyperacuity is of the same order as vernier acuity in human vision, which is capable of detecting a displacement in a line as small as 1/20th the diameter of a foveal cone, i.e. 1 arc second). Thus the ratio of improvement from these two mechanisms is

$$G_{p,g} = G_p \times G_g = 31.3 \times 32 = 1{,}002 \quad . \qquad (4\text{-}27)$$

We conclude therefore, that ***the use of log-polar coordinates compounded with discrete Gabor filter phase improves stereo resolution by a ratio of 1,000-to-1*** over conventional rasters with one-pixel resolution.

4.6 Gabor Filters in Log-Polar Coordinates

The 8x8 window depicted in figure 4-15 represents a small local neighborhood in Cartesian coordinates. What happens in log-polar coordinates? Consider first the fovea, which can be laid out in linear coordinates such as Cartesian or hexagonal as described in section 2. Hexagonal local geometry can be accommodated by affine transformation of Gabor filter coefficients, i.e. re-weighting the coordinates to account for the underlying hexangular areas superimposed on the square pixels of the Cartesian definition. Figure 4-16 schematically depicts a hexagonal arrangement of pixels with Gabor filter weighting superimposed.

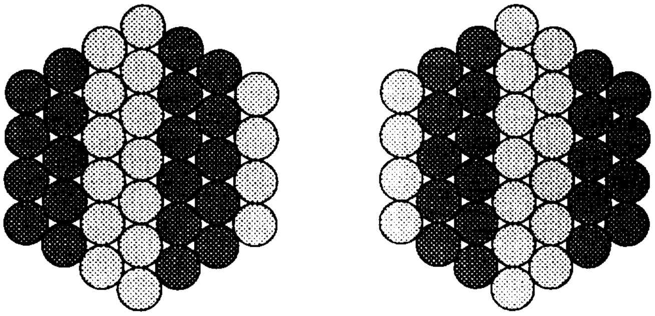


Figure 4-16. Hexagonal neighborhood Gabor filters

Figure 4-17 illustrates the two-pixel-wide striping pattern of "vertical" discrete Gabor filters superimposed on the hexagonal-to-polar blended fovea of figure 2-12. In the center of the fovea, geometry is linear and phase is proportional to image plane distance, as described in the preceding sub-section. In the periphery, however, columns are bowed along the ray-ring pattern of the pixels. Phase is to be interpreted here as linear *in log-polar coordinates*, i.e., within the "brain map" of figure 2-1b. Because the mapping is conformal, sufficiently small local neighborhoods are scaled, rotated versions of their counterparts in the pre-mapped (image plane) domain. That is, disparity measures must be scaled by local pixel size and oriented by local ray-ring layout.

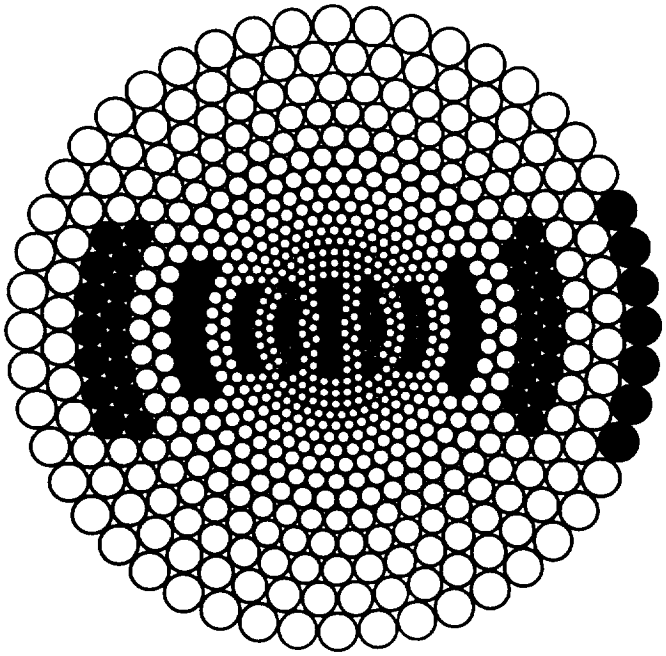


Figure 4-17. Discrete Gabor filter stripes for "vertical" orientation

Because the pixel pattern is intrinsically discrete, however, neighborhoods are *not* infinitesimal, and discrete Gabor filters in log-polar neighborhoods will exhibit some bowing distortion, proportionate to the coarseness of the ray sampling density, *n*. That is, for *n* rays, an 8x8 neighborhood arc subtends *16π/n* radians and bowing error (chord to arc distance) is

$$err_b = 1 - \cos\left(8\pi/_n\right) \tag{4-28}$$

as illustrated in figure 4-18.

Figure 4-19 plots equation 4-28, showing that relative error is 2% at $n = 128$, declining by a factor of 4 for every doubling thereafter, as one would expect from the Taylor series of the cosine. In designing a retina with pyramiding, one must plan for the decreased resolution at each subsequent level, which exacerbates bowing. For example, a 3-level pyramid with n = 512 at its finest level (close to the robot vision example of pages 3-28 and 4-17 where n = 503) has a coarsest level with n = 128, yielding a 2% bowing error according to figure 4-19. This value is quite manageable, corresponding to 5-6 bits of energy accuracy in filter output. However, log-polar retinas starting with, for example, n = 64 pixels per ring at the finest level of pyramid resolution, cannot support pyramiding because bowing will be excessive after the first level.

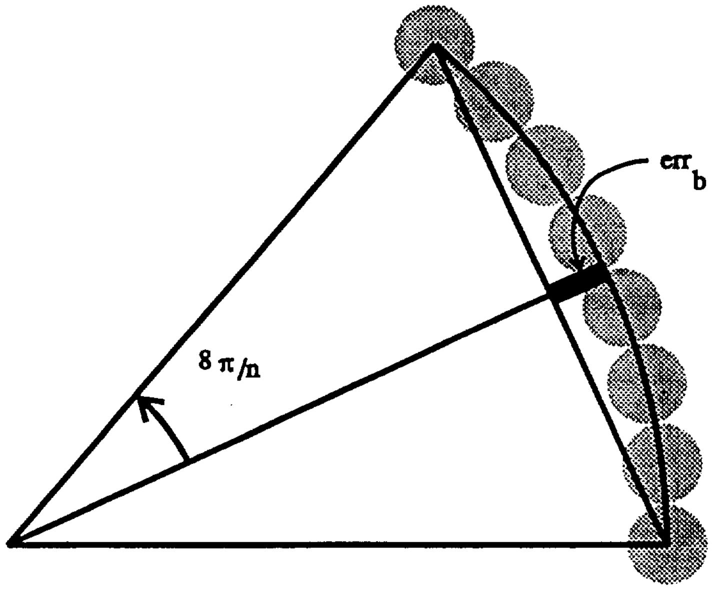


Figure 4-18. Bowed neighborhood

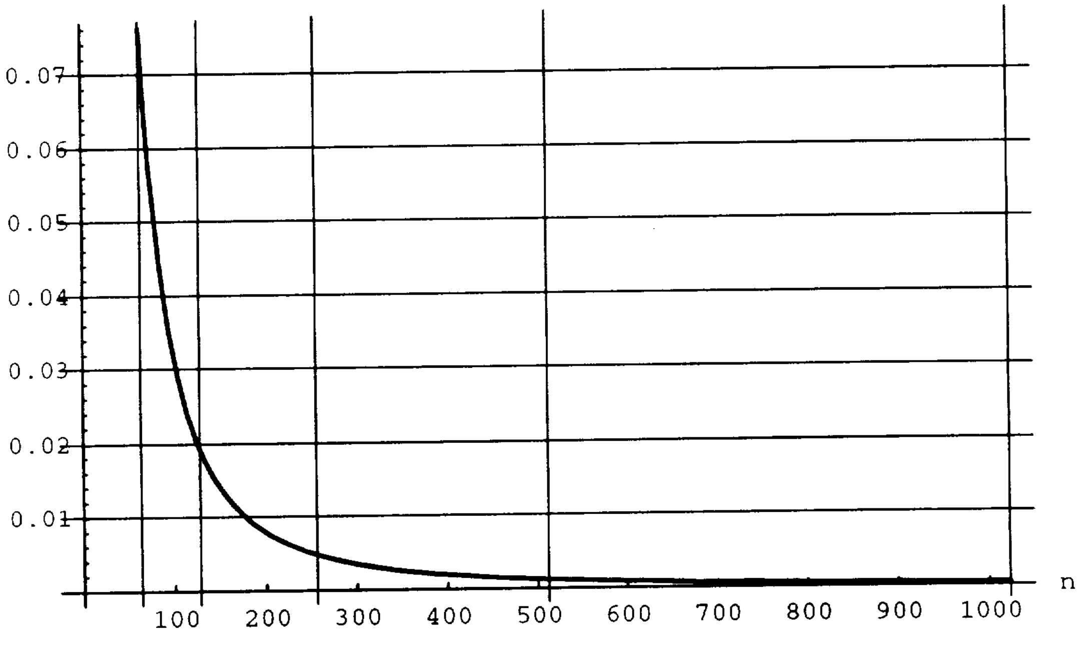


Figure 4-19. Relative bowing error from arcing window

In the periphery the curvilinear rings and rays of pixel infrastructure support curvilinear rays and rings of discrete Gabor filters. In the fovea, the linear pixel layout supports linear Gabor filter templates, albeit distorted where polar compromises increase near the periphery. Because the boundary between fovea and log-polar regions is seamless, the progression of discrete Gabor filters from nearest neighbor to nearest neighbor is also seamless as shown in figure 4-17.

However, conflicts arise if filter windows are chained sufficiently around an arc of the periphery to meet their "parallel" counterparts from the fovea. This is unavoidable because of the inherent difference in coordinate systems in the two domains. The conflict is of no consequence for stereo because disparity operators are inherently local. For binocular viewing, disparity is a 1-D phenomenon along the horopter and vertically oriented disparity operators are both necessary and sufficient to perform the task.

Global stereo surface reconstruction requires higher order integration of the local discrete Gabor filter outputs. The mechanisms of such integration are addressed by [Florack et al, 1992], but are beyond the scope of the current work.

Section 4 concludes the analysis section of this report. The next two sections describe equipment and experiments delivered to NASA to demonstrate some of the principles described above. The discrete Gabor filters were NOT implemented on the robot due to resource runout. However, C-programs were delivered to illustrate the phase behavior of the discrete Gabor function on step edges.

5.0 DESIGN OF MOBILE ROBOT WITH BINOCULAR VISION

A mobile robot with binocular vision, illustrated in figure 5-1, was constructed to test and demonstrate new methods for visually guided navigation. These methods are based on active binocular stereo which measures range quickly and accurately. The use of log-polar coordinates and Gabor filter feature detectors are important for efficient processing.

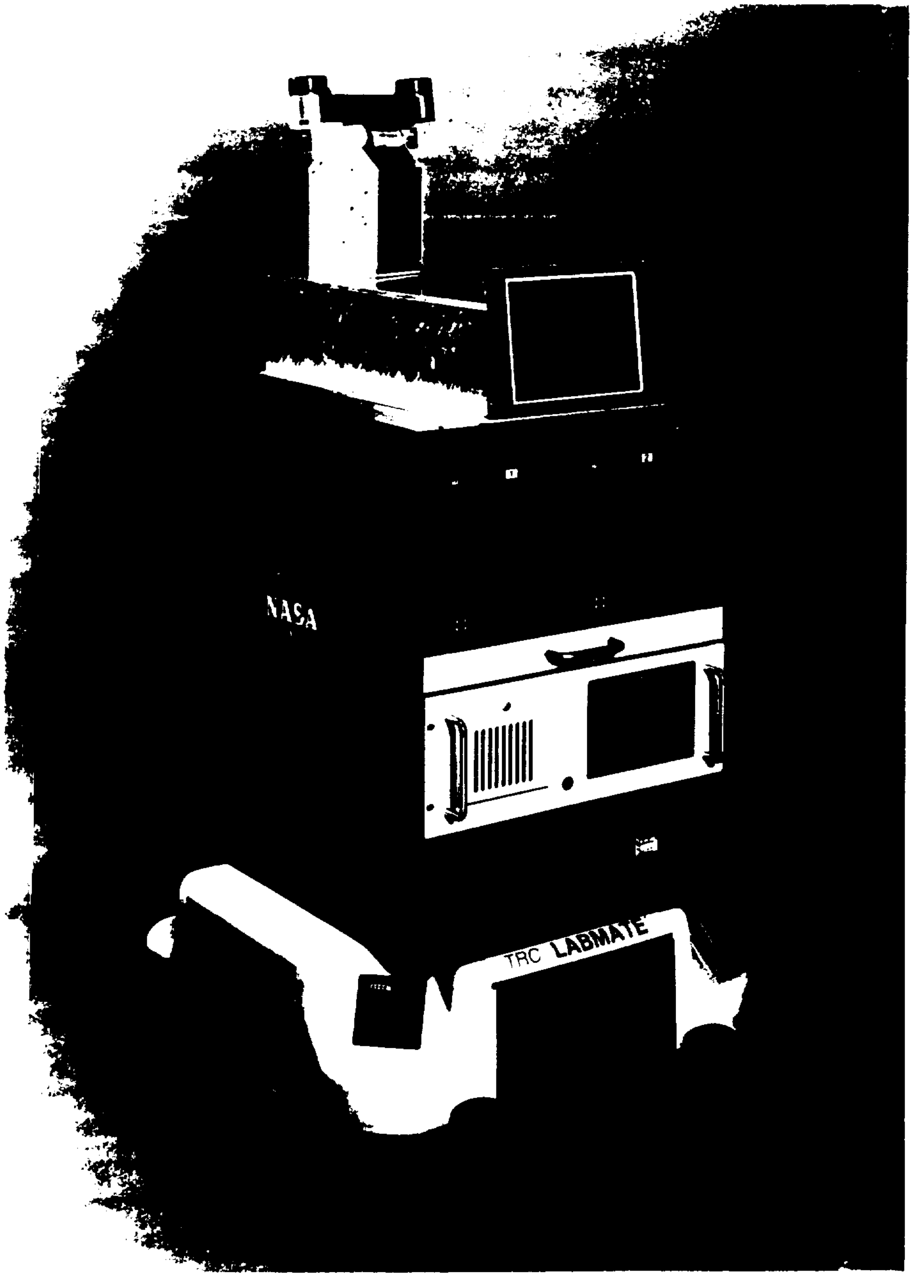


Figure 5-1. Mobile robot with binocular vision

Our major requirements for the robot were that it be self-contained, i.e. no umbilicals nor radio communication to external processing, and that visual processing be sufficiently fast to servo vehicle motion in real time. These objectives were achieved using off-the-shelf high-performance processing cards in an onboard PC/AT computer. The processing environment was designed for flexibility in research applications. Higher performance boards which become available in the future can replace current boards. Evolutionary migration to higher performance, or revolutionary replacement of the entire processing architecture can be achieved without modifying mechanical or electrical elements. Below we give an overview of the equipment. Details are given in manuals which accompany the equipment.

Figure 5-2 delineates the three major mechanical sections of the mobile robot. The top section contains the vision head, its drive motors and their amplifiers. The middle section contains video monitors, a PC/AT computer and associated boards, and batteries and power supplies for the electronics. The bottom section is a LabMate mobile robot base containing its own batteries and controllers. On the surface of the middle and bottom sections are sonar transducers which can be used for collision avoidance and navigation.

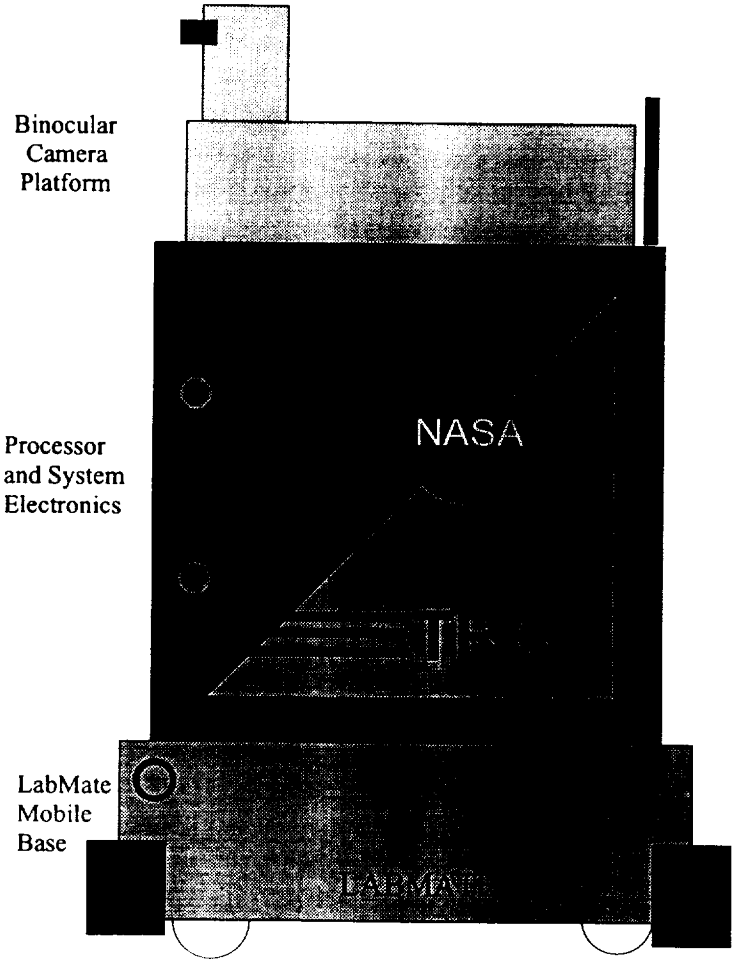
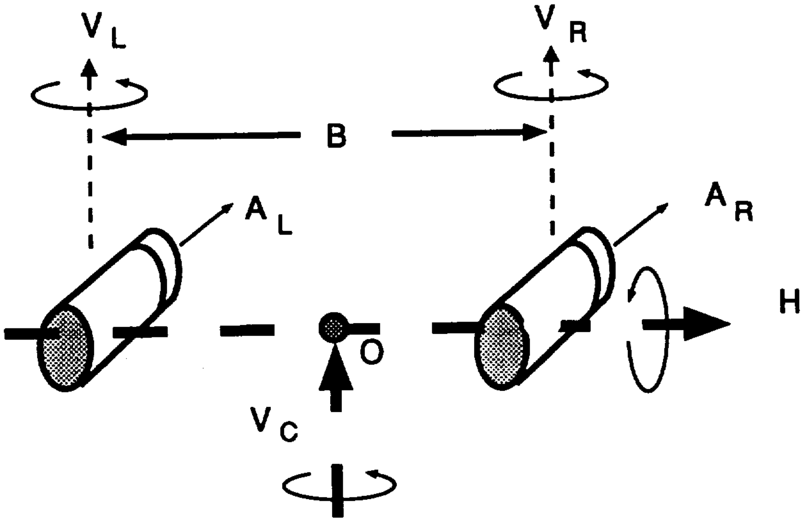


Figure 5-2. Side view of mobile robot

5.1 Binocular Camera Mount

The binocular vision head in the top section of the robot is a high-performance 4 degree-of-freedom camera mount holding a pair of microminiature Elmo cameras. Figure 5-3 schematically depicts the articulation and specifications. The outermost articulation consists of parallel independent vergence axes V_L and V_R driven directly by brushless DC motors. Speeds up to 1,000° per second are possible. The head tilt and pan axes, H and V_C are driven via stainless steel belt pulleys to eliminate backlash.



Axis	Range	Max Velocity	Max Acceleration
V_L , V_R	+/-45°	1,000°/sec	13,000°/sec^2
H	+/-45°	750°/sec	7,500°/sec^2
V_C	+/-45°	500°/sec	4,200°/sec^2

Figure 5-3. Binocular vision platform articulation and specs

Figure 5-4 illustrates the external structure of the camera platform. The base houses amplifiers and control electronics, and the motor for head pan drive about axis V_C . Figure 5-5 is an outline drawing with dimensions.

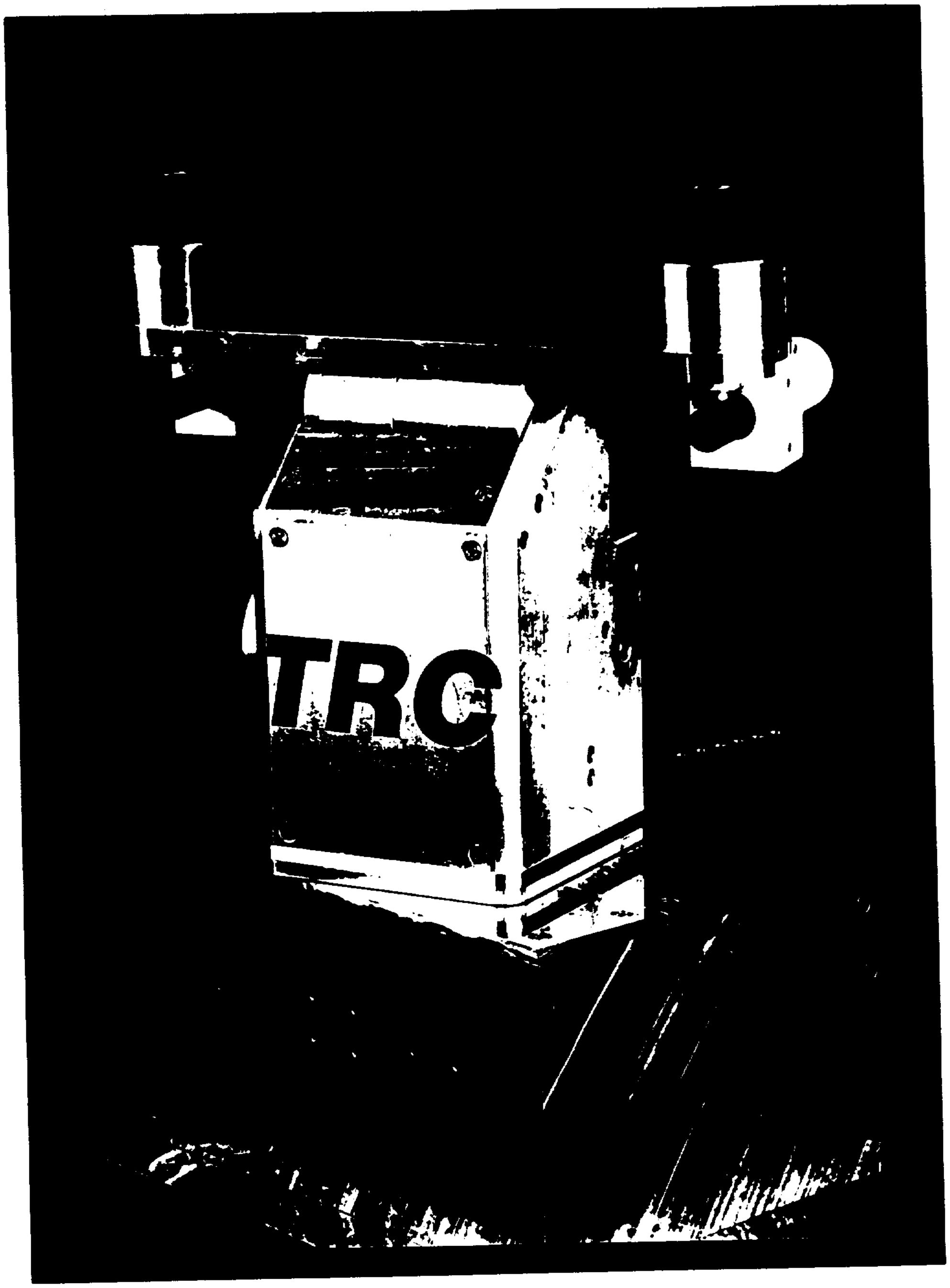


Figure 5-4. Photograph of binocular camera mount

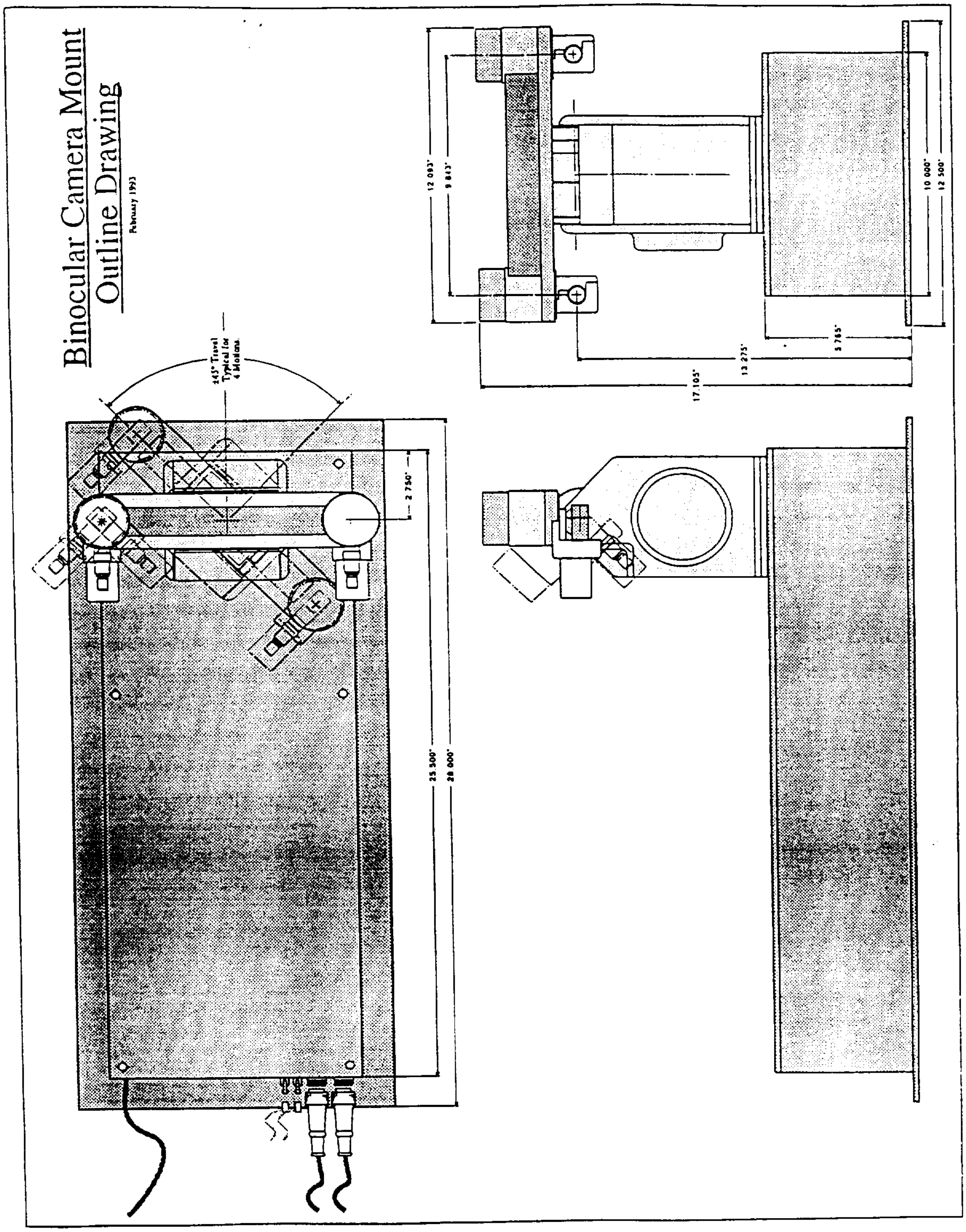


Figure 5-5. Outline drawing of binocular camera platform

5.2 Processor System Configuration

The processor is the major component of the middle section of the vehicle. Figure 5-7 depicts the system configuration and interconnects. An IBM PC/AT compatible is mounted in a 19" slide-out rack. Its slots are occupied by the 66 MHz 486 host, two Data Translation frame grabbers, two Alacron FT860/2-AT dual i860 RISC processor boards, and a Delta Tau PMAC 8-axis motor controller which drives the camera mount. The architecture of the Alacron boards is shown in figure 5-6. Note the high speed internal bus and dual RISC processors. The board is capable of 100 million instructions per second (MIPS). The Data Translation boards serve as image memory, and Alacrons as image processors for the two visual channels. Twin color 11" monitors are mounted above the 19" rack, to view the contents of image memory and processing results. These are very useful for illustrating visual processing in real time.

The keyboard for the computer slides out from the middle section; the monitor is a flat-panel VGA color screen anchored at the top of the middle section.The LabMate mobile robot base can be manually jockeyed via joystick, or driven by motion commands over an RS-232 port. The latter is the mode for autonomous navigation, whether by visual, sonar, or other sensory guidance.

Software for controlling the system is written in C. A menu driven user interface can be used to select various demos, described below. The modular software can be accessed via programming to implement custom designs at the discretion of the user.

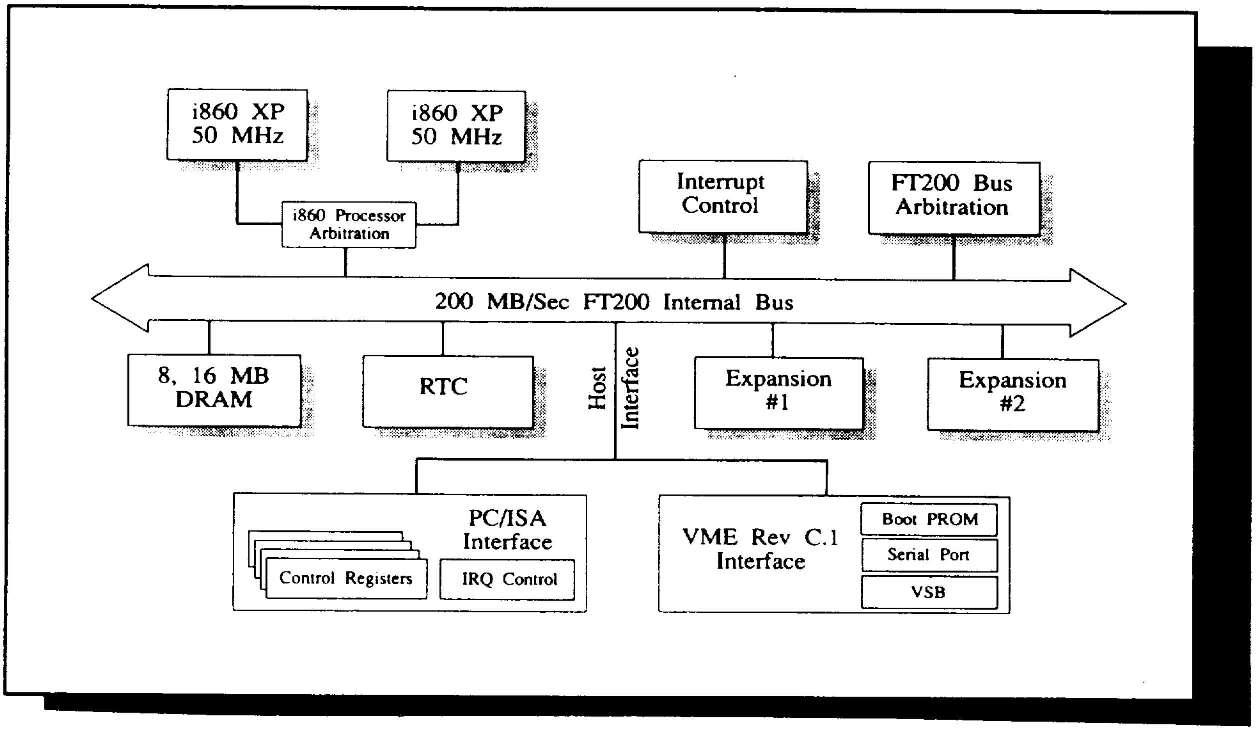


Figure 5-6. Alacron i860 board architecture

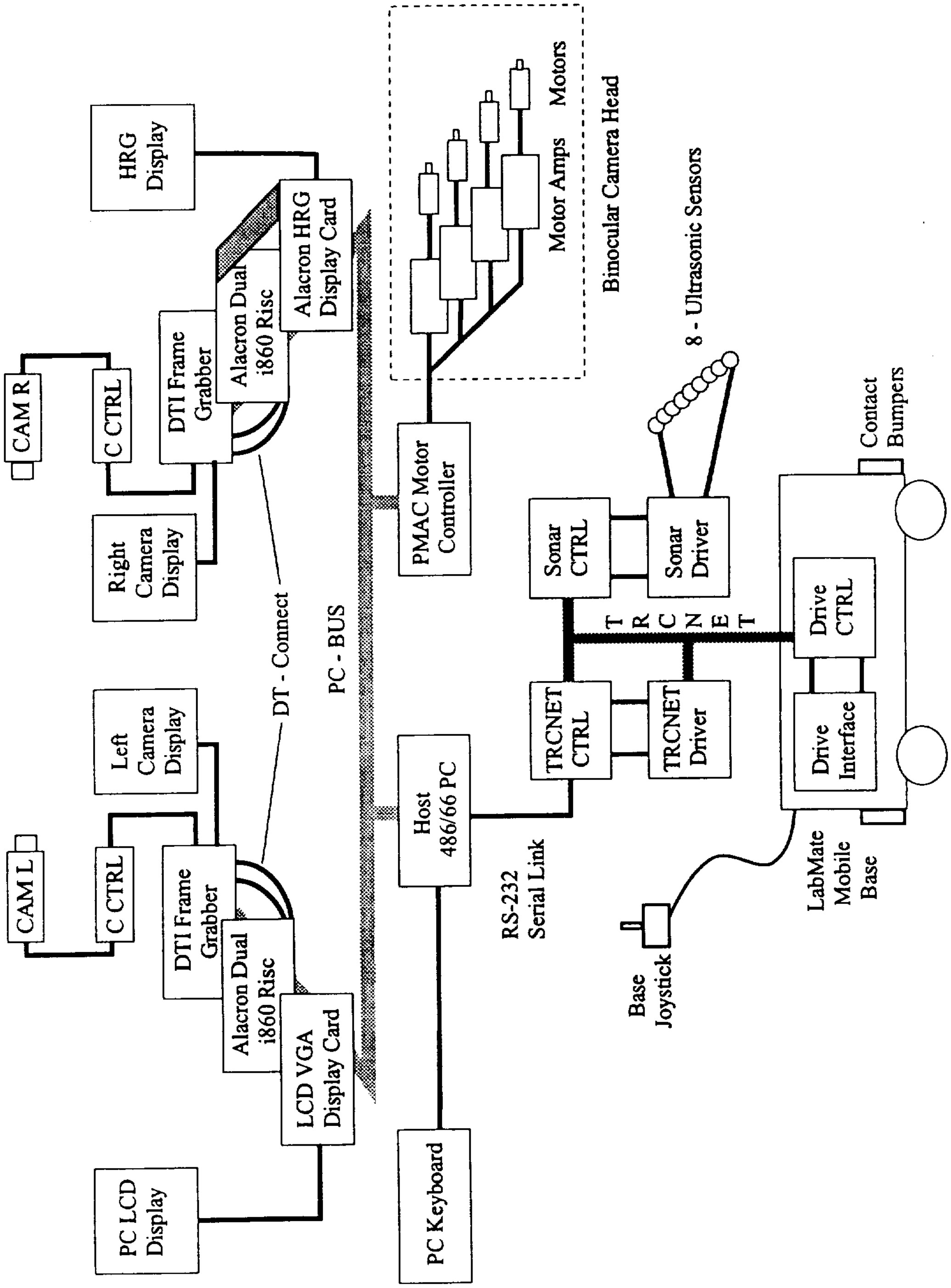


Figure 5-7. Mobile robot system configuration

6.0 CONTROL ALGORITHMS AND DEMONSTRATION SCENARIOS

Several scenarios were implemented to demonstrate the various elements of the system, and to provide software components which can be reconfigured to suit arbitrary experimental configurations which NASA may wish to develop without outside help. The major software elements are delegated to functions including image acquisition and routing, mapping, geometric computation, motion control, and user interfaces. Instructions for running the demonstrations, module description, and displays are given in the Robot and Applications Software Guide delivered with the hardware. We summarize the scenarios below.

6.1 Passive Scenarios

This group of demonstrations illustrates mathematical image processing involved in binocular stereo. Components include log-polar mapping, sub-pixel disparity measurements via the discrete Gabor filter, and 3-D inference from inverse projection through image and mechanical kinematic transformations.

Log-polar mapping

The first scenario is simply a demonstration of the log-polar mapping on the video monitor via menu commands. The resolution gradient is graphically demonstrated in a tiled reconstruction which displays live video from the camera with high resolution in the center, and low resolution in the periphery. An alternate configuration directly displays $(log(r), \theta)$ coordinates. While this view is difficult to intuit, it vividly demonstrates the data compression and warped streamline flow induced by motion, for example, rotation and zoom are transformed into vertical and horizontal translation, respectively.

The log-polar mapping and tiled reconstruction in these demonstrations are carried out in software on the Alacron i860 boards. The frame rate is about three per second, which is 20 times faster than the host could perform alone. This rate results from the concatenation of the following delays. Two video fields (16 milliseconds each) usually occur before a "ready-to-acquire" flag rises. This delay is unavoidable because host programs are asynchronous with respect to video clocks. Acquisition in the frame buffers requires two more field times (33 ms), and finally, transmission over the DT-connect bus to the Alacron processors requires two more field times (33 ms). Thus we have accumulated about 100 milliseconds for acquisition and transfer. Processing in the Alacrons requires nearly 250 milliseconds, for a grand total of 350 milliseconds.

A number of commercial products have emerged since the beginning of this project which could significantly speed up the mapping process. Pulnix now offers a low cost (under $2,500) digital camera which could dump data directly into Alacron memory, reducing the 100 milliseconds of acquisition to 33 milliseconds. Additionally, half the i860's are used for data transfer in the current configuration.

With the digital camera, Alacron computation capacity is therefore doubled, requiring only 125 milliseconds per frame for mapping. The total would thus be about 150 milliseconds, a frame rate of about 7 per second.

Even more significant gains could be realized by using the new PC/AT version of the CNAPS parallel processor from Adaptive Solutions, Inc. This unit has ten times the processing power of the Alacron, potentially cutting computation time to 15 milliseconds for a total of about 50 milliseconds per frame, which corresponds to 20 frames per second. This board was not yet available when components were purchased for the current system.

Demonstration of Subpixel Phase Sensitivity of Gabor Filter

A major accomplishment of this project was the design of a new, efficient, discrete Gabor filter which can be used for disparity measurements accurate to a fraction of a pixel. This demonstration, *filter.exe* residing in directory *c:\gabor\filter* in the on-board PC/AT, illustrates the robustness and sensitivity of the Gabor function in measuring the position of a synthetic visual edge to subpixel accuracy. The user can choose Gabor filter weights and orientation of an edge, then move the edge across an 8x8 pixel window in 1/8 pixel increments using the arrow key on the PC/AT keyboard. Defaults are provided for the parameters above if the user does not wish to think.

As the user moves the edge across the window, a display in the upper left shows the progress of the edge. Two sets of quadrature pair filters in the lower left depict progress over the weight masks likewise. To the right of each of these is a phase/magnitude plot which vividly illustrates the separation of pixel transit by phase quadrant. Mathematical explanation of this phenomenon is given in section 4 of this report and in [Weiman 94a].

Depth Ranging

In this scenario the camera drives are locked in an *arbitrary* position, and a point light source target is presented in the common area of their fields of view. The positions of the images of the light in left and right cameras determine ψ_l , ψ_r and φ_l , φ_r , respectively the horizontal and vertical components of the angles of deviation from the optical axes A_L and A_R in figure 3-2. These angles, together with the joint angles α_r, α_l , β , γ of the cameras, are fed in to the inverse kinematic equations (shown in detail in section 3.1 of this report) of the mechanism to determine 3-D position of the target.

The display on the PC/AT monitor shows a top (left half of screen) and side (right half of screen) view of the inferred depth. The camera is depicted as a large filled square, and the target as a small filled square. The display is dynamic; as the operator moves a target around in front of the cameras, the display moves in real time, tracking the target.

6.2 Active Scenarios

This group of scenarios illustrates active vision at progressively more complex levels. The lowest is stepwise camera tracking of a target. Next is continuous binocular tracking, and third is mobile robot vehicle control by visually following a moving target and maintaining a fixed following distance.

Saccades

In biology, a *saccade* is a sudden jump in the direction of gaze of the eye, from one stationary position to another some distance away. The move is characterized by maximum accelleration and rapid settling time. Vision is disabled during the move, whose purpose is to attend to a new point of interest as rapidly as possible.

In the demonstration scenario, the cameras are stationary in some arbitrary initial position. A light is held at some general position within the fields of view of both cameras. When the "enter" key is depressed, the positions of the images of the lights are fed through the inverse kinematics to compute the motor commands necessary to bring the images to the center of the field of view. The cameras then move to that position via "open loop" control (as in the biological counterpart) as quickly as possible. Successful centering verifies the kinematic model. This motion control mechanism can be used as a building block in more sophisticated artificial vision scenarios where the peripheral stimulus might be motion, color, or perceptual salient.

Camera Target Tracking

In this demonstration, the target is moved about in front of the cameras, which dynamically track it. The tracking algorithm uses velocity control proportional to angular distance from image center to target. Thus, higher speeds occur for greater deviations. The cascaded articulation is handled as follows. Vergence cameras move to zero out the horizontal component of angle between target image and optical axis. Independently, pan does likewise for the vertical component of target image. Beneath these, the head pan axis tries to zero out the average angle of vergence cameras, without reference to image position, i.e., based only on encoder position. The behavior in response to a sudden jump in target position is saccadic motion of vergence and tilt axes, and slower "catching up" of the head pan axis to squarely address the target. This is much like primate visual response [Ballard, 1991], [Brown, 1990a, b]. The eyes instantly flick to a new target while the more massive head catches up. For tracking a smoothly moving target, this simple algorithm provides smooth camera motions. Oscillatory coupling between head-pan and vergence is eliminated by the longer effective time constant of the former.

We formalize the preceding description as follows. Let α_l and α_r refer to angular position of the vergence motors about the V_L and V_R axes in figure 3-2, let β refer to tilt angle about the H axis, and γ to head pan angle about the V_C axis.

Let ψ_l and ψ_r refer to the horizontal components of the angle between target and optical axes A_L and A_R in the left and right camera images, and φ_l and φ_r the vertical components. Figure 6-1 illustrates the control algorithm described in the preceding paragraph. Note that vergence and tilt are velocity outputs controlled by *image* position error, whereas neck pan is purely a function of summed vergence position from encoders.

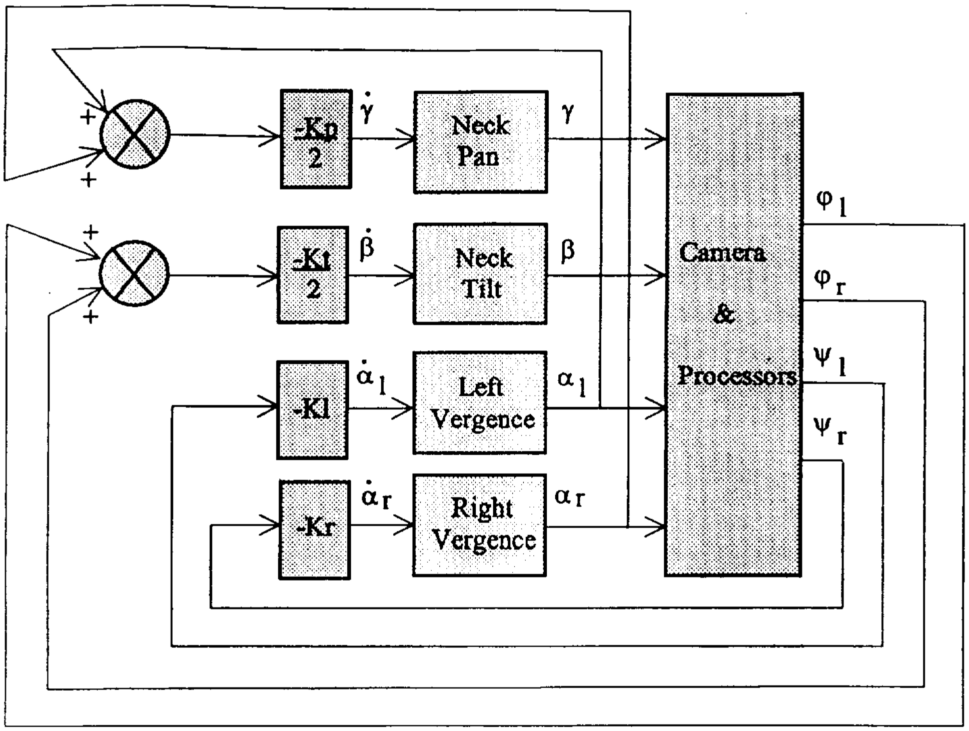
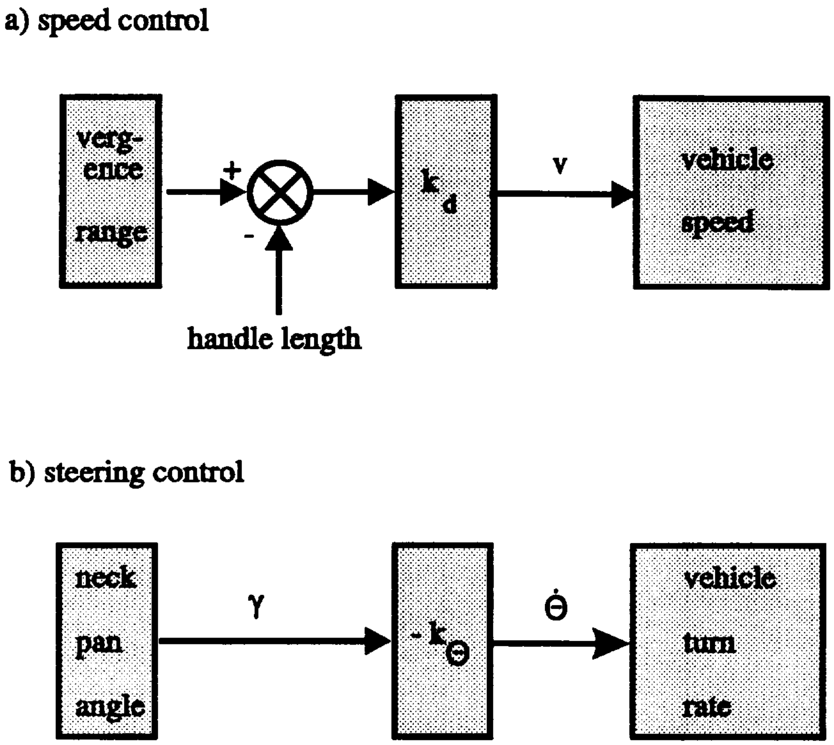


Figure 6-1. Tracking control for 4-DOF camera motion platform

Integrated Vehicle and Camera Tracking

Integrated vehicle and camera control is a simple extension of the preceding camera motion servo. The vehicle, which supports the binocular camera mount, is driven to exercise "follow-me" behavior, as if pulled by an imaginary stretchy handle reaching from the target position to the center of the vehicle. This is accomplished by specifying the length of the "handle" *a priori* and defining the distance error as the difference between this length and the distance to the target as measured by vergence angle of the cameras, which are tracking the target as specified in figure 6-1. This error signal controls the forward velocity of the robot. Thus, the handle acts like a spring, stretching to speed the vehicle, and compressing to slow it down. Figure 6-2a illustrates the system diagram.

Vehicle steering is controlled in the same manner as the camera mount neck. The error signal for vehicle steering control is neck pan angle γ, as illustrated in figure 6-2b. This signal controls heading turn rate by adjusting the speed of left and right drive wheels equally in opposite directions.



a) speed control

b) steering control

Figure 6-2. Vehicle control based on inputs from camera mount position

The servo algorithms described above are examples which have been programmed into the deliverable vehicle. These programs may be used as modules for more complex behaviors, programmable in C. The open architecture and modern software development environment give the user complete control over all algorithms.

7.0 SUMMARY AND REFERENCES

We have presented design principles, parameters, and a prototype platform for an active binocular vision system which operates in real time using innovative techniques at several fundamental levels. At the sensor level we prescribe log-polar coordinates to maximize central resolution, minimize pixel count, and provide useful coordinates for geometric transformations. Image plane pixel count is reduced by 30-to-1 compared to uniform tessellations with equivalent central resolution and field of view. At the first processing level, we recommend discrete Gabor filters as 2-D DSP's to extract phase information at sub-pixel resolutions. The compact neighborhoods and simple structure of these filters is commended for VLSI implementation in highly parallel form. ***Together, the log-polar pixel layout and Gabor filter phase disparity improve binocular stereo range resolution by a factor of 1,000-to-1 compared to conventional Cartesian rasters of comparable pixel count, field of view, and single pixel resolution.***

Finally, at the mechanical level, we constructed a high precision binocular camera platform with no backlash, employing micro-miniature cameras for accurate, high-speed motion. The controls and processors are commercially available PC/AT based boards for flexibility and ease of programming. The system is housed on a self-contained mobile robot without umbilicals. The integrated system can serve as a testbed for binocular vision research in realtime scenarios.

References

Aloimonos, Y. "Purposive and Qualitative Active Vision", *10th International Conference on Pattern Recognition,* IEEE, pp 346-360, June 1990.

Aloimonos, Y., Weiss, I., and Bandopadhay, A., "Active Vision", *First International Conference on Computer Vision 1986,* IEEE , pp. 35-54, 1987.

Bajcsy, R., "Active Perception", *Proceedings of the IEEE,* Volume 76, pages. 996-1005, August, 1988.

Ballard, D. H., "Reference Frames for Animate Vision", *International Joint Conference on AI,* AAAI, 1989.

Ballard, D. H., "Animate Vision", *Artificial Intelligence Conference 48,* pp. 57-86, Elsevier, 1991.

Bishay, M., Kara, A., Wilkers, D. M., Peters, R. A., and Kawamura, K. "An Active Vision Approach for Locating Salient Features of Objects Using Log-Polar Mapping with No Camera Motion", *Vision Interface '94,* pp. 64-72, 1994.

Braithwaite, R. N., and Beddoes, M. P. "Disparity Estimation and Direct Passive Navigation using Stereo Gabor Filters", *Proceedings of the Vision Interface Conference,* Banff, Canada, pp. 56-63, May, 1994.

Brown, C. M., “Gaze controls cooperating through prediction”, *Image and Vision Computing*, Vol 8, No. 1, pp. 10-17, February 1990a.

Brown, C. M., “Gaze controls with interactions and delays”, *IEEE Trans. on systems, Man, and Cybernetics*, Vol. 20, No. 3, May 1990b.

Burt, P. J., and Adelson, E. H., “The LaPlacian pyramid as a compact image code”, *IEEE Trans. Comm.*, Vol. COM-31, April 1983.

Burt, P. J., “Smart sensing with a pyramid vision machine”, *Proc IEEE*, Vol 76, pp.1006-1014, August, 1988.

Chaikin, G. M., and Weiman, C.F.R., "Conformal Computational Geometry for Machine Vision", *Proc. of the Fifth International Conf. on Pattern Recognition*, pp. 1106-1110, Miami, December 1980,

Chaikin, G. M. and Weiman, C. F. R., "Image Processing System", U. S. Patent 4,267,573, May 12, 1981 (log-polar graphics and image processing method).

Coombs, D. J., Real-time Gaze Holding in Binocular Robot Vision, Ph. D. Dissertation, Dept. of Computer Science, University of Rochester, January, 1992.

Crowley, J. L., Bobet, P., and Mesrabi, M., “Gaze Control for a Binocular Camera Head”, *Second European Conference on Computer Vision*, pp. 588-596, May, 1992

Daugman, J. C., "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters", *Journal of the Optical Society of America*, Vol 2, No. 7, pp. 1160-1169, July 1985.

Fisher, T. E., and R. D. Juday, "A programmable video image remapper", Proc. SPIE Conf. on Pattern Recognition and Signal Processing, Vol. 938, *Digital and Optical Shape Representation and Pattern Recognition*, pp. 122-128, Orlando, 1988.

Fleet, D. J., Measurement of Image Velocity, Kluwer Academic Publishers, Boston, 1992.

Fleet, D. J., and Jepson, A. D., “Stability of phase information”, *IEEE Trans PAMI*, Vol 15, pp. 1253-1268, December, 1993.

Florack, L.M.J., Romeny, B. M. tH., Koenderink, J.J., and Viergiver, M. A., “Scale and the differential structure of images”, *Image and Vision Computing*, Vol 10, pp 376-387, 1992.

Freeman, W. T., and Adelson, E. H., “The design and use of steerable filters”, *IEEE Trans. PAMI*, Vol 13, No. 9, pp. 891-906, September, 1991.

Gabor, D., "Theory of Communication", *Journal of the Inst. of Elec. Engrs*, v. 93, pp. 429-459, 1946.

Gennert, M. A. and Malin, J. A., "Stereo vision using Gabor receptive fields", *Proceedings of SPIE Conference on Intelligent Robots and Computer Vision XI: Biological, Neural Net and 3-D Methods,* Casasent, Vol 1826, pp 64-75, Boston, November 1992.

Griswold, N. C., Lee, J.S., and Weiman, C.F.R., "Binocular Fusion Revisited Utilizing a Log-Polar Tessellation", pp. 421-457 in Computer Vision and Image Processing, Academic Press, 1992.

Horn, B. K. P., and Schunck, B. G., "Determining Optical Flow", *Artificial Intelligence*, Vol. 17, No. 1-3, pp. 185-204.

Huebel, D. G. and Wiesel, T. N. "Receptive fields, binocular interaction, and functional architecture in the cat's visual cortex", *Journal of Physiology*, Vol 160, pp. 106-154, 1962.

Jain, R., Bartlett, S. L., and O'Brien, N., "Motion Stereo Using Ego-Motion Complex Logarithmic Mapping", *IEEE Trans. PAMI*, Vol 9, No. 3, pp. 356-369, May, 1987.

Julesz, B., Foundations of Cyclopean Perception, Univ. of Chicago Press, 1971.

Kanade T. and Okutomi, M., "A Stereo Matching Algorithm with an Adaptive Window: Theory and Experiment", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol 16, pp. 920-932, September, 1994.

King, S. J. and Weiman, C.F.R., "HelpMate Autonomous Mobile Robot Navigation System", *SPIE Vol. 1388 - Mobile Robots V*, 1990a.

Koenderink, J. J., and van Doorn, "Representation of Local Geometry in the Visual System", *Biological Cybernetics*, Vol 55, pp. 367-375, 1987.

Koenderink, J. J., "Operational Significance of Receptive Field Assemblies", *Biological Cybernetics*, Vol 58, pp. 163-171, 1988.

Kuglin, C. D., and Hines, D. C., "The Phase Correlation Image Alignment Method", *Proceedings of the 1975 International Conf. on Cybernetics and Society*, pp. 163-165, September 1975.

Laine, A. and Fan, J., "Texture classification by Wavelet Packet Signatures", *IEEE Transactions on PAMI*, Vol. 15, No. 11, pp. 1186-1191, November, 1993.

Laine, A., Schuler, S., and Girish, V., "Orthonormal wavelet representations for recognizing complex notations", *Machine Vision and Applications*, Vol 6, pp. 110-193, 1993.

Lumelsky, V. J., "Detection of Mutual Interference in Multirobot Systems", Tech Report 8508, Dept of EE, Yale University, May, 1985.

Marr, D. and Poggio, T., "A Computational Theory of Human Stereo Vision", *Proceedings of the Royal Society, London B*, Vol 204, pp. 301-328, 1979.

Mead, C. and Ismail, M., Analog Implementations of Neural Networks, Kluwer Publishing, 1989.

Meyer, Y. (translated by R. D. Ryan), Wavelets: Algorithms and Applications, Society for Industrial and Applied Mathematics, Philadelphia, 1993.

Mundi, J. L., and Zisserman, A., Geometric Invariance in Computer Vision, MIT Press, Cambridge, 1992.

Nishihara, H. K., "Practical real-time imaging stereo matcher", *Optical Engineering*, Vol. 23, pp. 536-545, October, 1984.

Nishihara, H. K., "Tests of a sign correlation model for binocular stereo", *Investigative Ophthalmology and Visual Science*, Vol. 30, pp. 389, 1989.

Ohzawa, I., DeAngelis, G. C., and Freeman, R. D., "Stereoscopic depth discrimination in the visual cortex: neurons ideally suited as disparity detectors", *Science*, Vol 249, pp. 1037-1041, August, 1990.

Oppenheim, A. V. and Lim, J. S., "The Importance of Phase in Signals", *Proc. IEEE*, Vol. 69, No. 5, pp. 529-541, May 1981.

Oster, G., The Science of Moiré Patterns, Edmund Scientific Company, Barrington, New Jersey, 1965.

Pahlavan, K., Eklundh, J.-O., and Uhlin, T., "Integrating Primary Ocular Processes", *Second European Conference on Computer Vision*, Springer, pp. 526-541, May, 1992.

Patterson, R. and Martin, W. L., "Human Stereopsis", *Human Factors*, Vol 34, pp. 669-692, 1992.

Pentland, A. P., "Linear shape from shading", *International Journal of Computer Vision*, Vol 4, pp. 153-162, 1990.

Pollen, D. and Ronner, S, "Phase relationships between adjacent simple cells in the visual cortex", *Science*, Vol 212, pp 1409-1411, 1982.

Rojer, A. S. and Schwartz, E. L., “Design Considerations for a Space-Variant Sensor with Complex Logarithmic Geometry”, *Proc. 10th International Conference on Pattern Recognition*, Atlantic City, pp. 278-285, 1990.

Sandini, G., and P. Dario, "Active vision based on space-variant sensing”, in *Proceedings of the 5th International Symposium on Robotics Research*, Tokyo, MIT Press, 1989.

Sandini, G., Tagliosco, V., and Tistarelli, M.,"Analysis of Object Motion and Camera Motion in Real Scenes", *Proceedings of the 1986 IEEE International Conf. on Robotics*, pp. 627-633, 1986.

Sanger, T. D., “Stereo disparity computation using Gabor filters”, *Biological Cybernetics*, Vol 59, pp. 405-418, 1988.

Schiehlen, J. and Dickmanns, E. D., “Design and Control of a Camera Platform for Machine Vision”, *Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems (IROS ‘94)*, Munich, Germany, pp. 2058-2063, September 1994.

Schultze, M., "Zur Anatomie und Physiologie der Retina", *Archiv für Mikroskopische Anatomie*, Vol 2, pp. 175-286, 1866.

Schwartz, E. L., "Computational Anatomy and Functional Architecture of Striate Cortex: A Spatial Mapping Approach to Coding", *Vision Research*, Vol 20, pp. 645-669, 1980.

Schwartz, E. L., "On the Mathematical Structure of the Retinotopic Mapping of Primate Striate Cortex", *Science*, Vol 277, page 1066, 1985.

Theimer, W. M., Mallot, H. A., and Tölg, S., "Phase Method for binocular vergence control and depth reconstruction", *Proceedings of of SPIE Conference on Intelligent Robots and Computer Vision XI: Biological, Neural Net and 3-D Methods,* Casasent, Vol 1826, pp 76-87, Boston, November 1992.

Tölg, S., “Gaze control for an active camera system by modeling human pursuit eye movements”, *Proceedings of of SPIE Conference on Intelligent Robots and Computer Vision XI: Algorithms, Techniques and Active Vision,* Casasent, Vol 1825, pp 585-598, Boston, November 1992.

Van der Spiegel, J., G. Kreider, C. Claeys, I Debusschere, G. Sandini, P. Dario, F. Fantini, P. Bellutti, G. Soncini , "A Foveated Retina-Like Sensor Using CCD Technology", in Analog VLSI and Neural Network Implementations, C. Mead and M. Ismail, Eds., DeKluwer Pubs, Boston, 1989.

Wallace, R. S., Ong, P-W., Bederson, B.B., and Schwartz, E. L., "Connectivity Graphs in Space Variant Active Vision", Tech Report VAI-1, Vision Applications, Inc., 1991.

Weiman, C. F. R. and G. Chaikin, "Logarithmic Spiral Grids for Image Processing and Display", *Computer Graphics and Image Processing*, (11), pp. 197-226, 1979.

Weiman, C. F. R., "Exponential sensor array geometry and simulation", *Proc. SPIE Conf. on Pattern Recognition and Signal Processing*, VoL 938, *Digital and Optical Shape Representation and Pattern Recognition*, pp. 129-137, Orlando, 1988a.

Weiman, C. F. R., "3-D sensing with exponential sensor arrays", *Proc. SPIE Conf. on Pattern Recognition and Signal Processing*, VoL 938, *Digital and Optical Shape Representation and Pattern Recognition*, Orlando, 1988b.

Weiman, C. F. R. and Juday, R. D., "Tracking Algorithms Using Log-Polar Mapped Image Coordinates", *Proc. SPIE Conf. on Intelligent Robots and Computer Vision VIII: Algorithms and Techniques*, VoL 1192, Phila., November 1989.

Weiman, C. F. R., "Polar Exponential Sensor Arrays Unify Iconic and Hough Space Representation", *Proc. SPIE Conf. on Intelligent Robots and Computer Vision VIII: Algorithms and Techniques*, VoL 1192, Phila., November 1989b.

Weiman, C.F.R., Robot Vehicle Video Image Compression, Phase I SBIR final report, TACOM contract DAAE07-88-C-RO68 to Transitions Research Corporation, 1989c.

Weiman, C. F. R., "Log Polar Vision for Mobile Robot Navigation", *Electronics Imaging: International Electronic Imaging Exposition and Conference*, pp.382-386, Oct 1990a.

Weiman, C. F. R., "Video Compression Via Log Polar Mapping", *Proc. SPIE Conf. on Real-Time Image Processing II*, VoL 1295, Orlando, April 1990b.

Weiman, C. F. R., "Efficient Discrete Gabor Functions for Robot Vision", *Proc. SPIE Conf. on Wavelet Applications*, VoL 2242, Orlando, April, 1994a.

Weiman, C. F. R., "Video Compression by Matching Human Perceptual Channels", *Proc. SPIE Conf. on Visual Information Processing III*, VoL 2239, Orlando, April, 1994b.

Weiman, C. F. R., and Evans, J.M., Digital Image Compression Employing a Resolution Gradient, U.S. Patent No. 5,103,306, 1992.

Yeshurun, T., and Schwartz, E. L., "Cepstral Filtering on a Columnar Image Architecture: A Fast Algorithm for Binocular Stereo Segmentation", *IEEE Trans. PAMI*, Vol 11, no 7, November, 1989.

Yeshurun, T., and Schwartz, E. L., "Shape Description With a Space Variant Sensor: Algorithms for Scan-Path, Fusion, and Convergence Over Multiple Scans", *IEEE Trans. PAMI*, VoL 11, no. 11, pp. 1217-1222, November, 1989.